Document Page Layout Analysis

Bhabatosh Chanda
Electronics and Communication Sciences Unit
Indian Statistical Institute
Kolkata 700108, India

Acknowledgement

- Amit Das, IIEST, Sibpur
- Sekhar Mandal, IIEST, Sibpur
- Sanjoy Kumar Saha, Jadavpur Univeristy
- Ranjan Mandal, Indian Statistical Institute

Outline

- Introduction
- Projection method
  - Zone content classification
- Morphological operators
  - Skew correction
- Morphology based method
- Deep learning based method
- Performance evaluation
- Database: examples
- Conclusion

Introduction

- Problem description
- Motivation
  - Improve performance of OCR
  - Data compression
  - Graphics recognition
  - Browsing and navigation
- Physical and logical structure

Problem Description

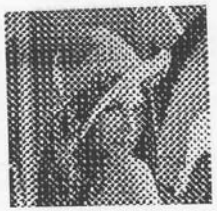


Table 1: Example of table

Roll	Lit.	Sci.	Soc.	Math.
1	38	95	67	85
23	67	83	19	78
43	17	93	13	92
67	19	16	34	56

Figure 1: Example of figure inclusion.

This page is used to develop and algorithm for document page image segmentation. Page should contain mathematical equations, tables, section headings, graphics or line drawing and half-tone images alongwith regular text. The page is generated using LaTex word processing system. Then the page is printed on a HP-5MP laserjet printer, followed by scanned in by HP 4C scanner. Scanned in image may have tilt.

1 Math equations

Below we give an example of mathematical equation computing mean of a set of data points followed by the formula for computing variances of the same. Equations with or without equation number are given.

$$\overline{x} = \frac{1}{n}\sum_{i=0}^{n} x_i$$

Following one is an example of equation array with equation number. However, other style of equations are not considered here because of lack of space. So the following equation is the example of the second type that is equation with number.

$$\begin{aligned} \sigma_x &= \frac{1}{n}\sum_{i=0}^{n}(x_i - \overline{x})^2 \\ &= \frac{1}{n}\sum_{i=0}^{n} x_i^2 - \overline{x}^2 \end{aligned} \quad (1)$$

Now we give an example of table that has column and row separators as well as captions and number. Finally we put a line drawing which is basically a normal probability density function. Hence this page illustrates all the basic entities we like to deal with in a document page image segmentation. Here is the end of this page.

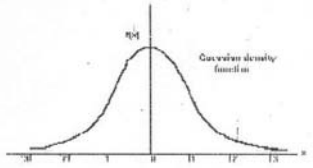


Figure 2: Example of Graphics inclusion.

Thanks for your attention. Resolution of scanning is 100 dpi (i.e., dots per inch).

Objective

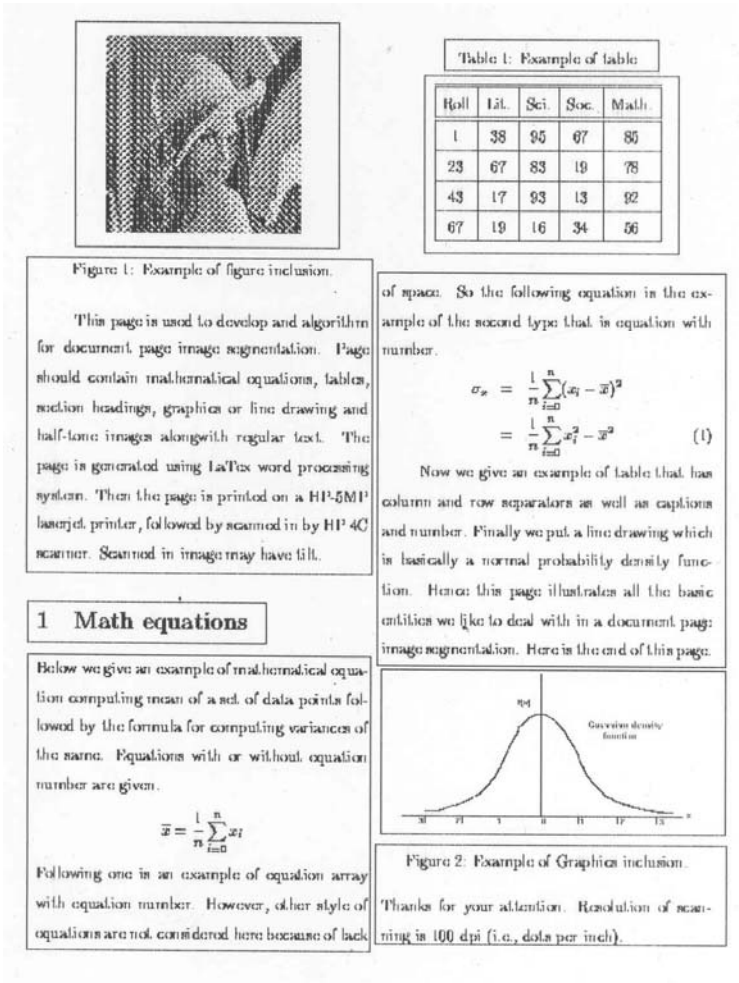


Table 1: Example of table

Roll	Lit.	Sci.	Soc.	Math
1	38	95	67	85
23	67	83	19	78
43	17	93	13	92
67	19	16	34	56

Figure 1: Example of figure inclusion.

This page is used to develop and algorithm for document page image segmentation. Page should contain mathematical equations, tables, section headings, graphics or line drawing and half-tone images alongwith regular text. The page is generated using LaTex word processing system. Then the page is printed on a HP-5MP laserjet printer, followed by scanned in by HP 4C scanner. Scanned in image may have tilt.

1 Math equations

Below we give an example of mathematical equation computing mean of a set of data points followed by the formula for computing variances of the same. Equations with or without equation number are given.

$$\overline{x} = \frac{1}{n}\sum_{i=0}^{n} x_i$$

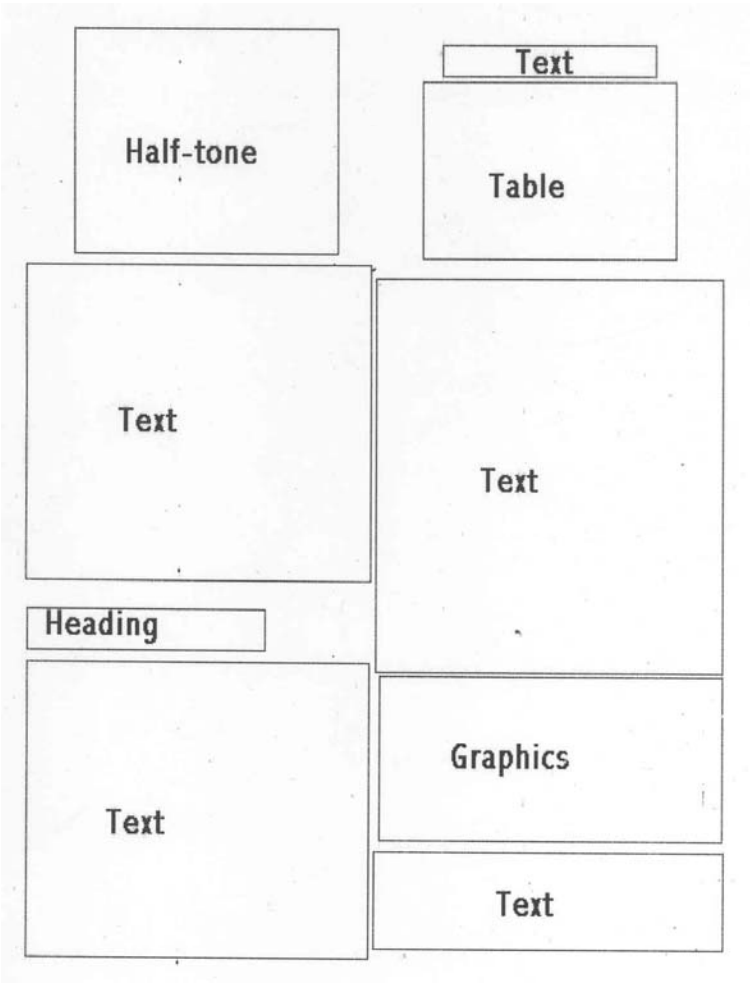
Following one is an example of equation array with equation number. However, other style of equations are not considered here because of lack of space. So the following equation is the example of the second type that is equation with number.

$$\begin{aligned}\sigma_x &= \frac{1}{n}\sum_{i=0}^{n}(x_i - \overline{x})^2 \\ &= \frac{1}{n}\sum_{i=0}^{n} x_i^2 - \overline{x}^2 \end{aligned} \quad (1)$$

Now we give an example of table that has column and row separators as well as captions and number. Finally we put a line drawing which is basically a normal probability density function. Hence this page illustrates all the basic entities we like to deal with in a document page image segmentation. Here is the end of this page.

Figure 2: Example of Graphics inclusion.

Thanks for your attention. Resolution of scanning is 100 dpi (i.e., dots per inch).



Major Source of Document Pages

1. Books
2. Journals
3. Magazines
4. Newspapers
5. Forms and leaflets
6. Reports

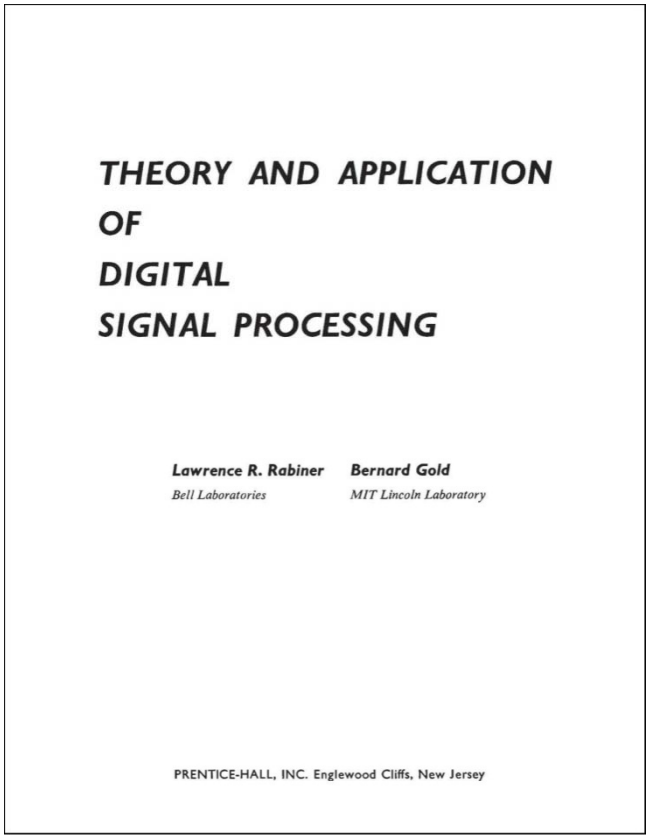
Types of document pages

Consider books and journals

- Title page
- Publisher's page
- Table of Contents
- Text page
- Index page

Different types of pages

Title page



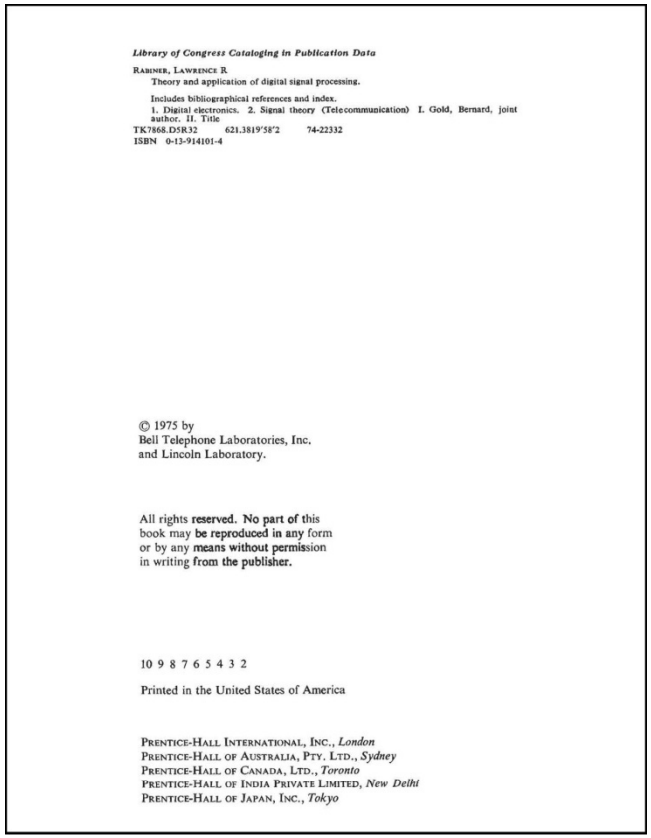
THEORY AND APPLICATION
OF
DIGITAL
SIGNAL PROCESSING

Lawrence R. Rabiner — *Bell Laboratories*

Bernard Gold — *MIT Lincoln Laboratory*

PRENTICE-HALL, INC. Englewood Cliffs, New Jersey

Publisher's page



Library of Congress Cataloging in Publication Data

RABINER, LAWRENCE R

Theory and application of digital signal processing.

Includes bibliographical references and index.

1. Digital electronics. 2. Signal theory (Telecommunication) I. Gold, Bernard, joint author. II. Title

TK7868.D5R32 621.3819'58'2 74-22332

ISBN 0-13-914101-4

10 9 8 7 6 5 4 3 2

Printed in the United States of America

PRENTICE-HALL INTERNATIONAL, INC., *London*
PRENTICE-HALL OF AUSTRALIA, PTY. LTD., *Sydney*
PRENTICE-HALL OF CANADA, LTD., *Toronto*
PRENTICE-HALL OF INDIA PRIVATE LIMITED, *New Delhi*
PRENTICE-HALL OF JAPAN, INC., *Tokyo*

Different types of pages

Table of Content page

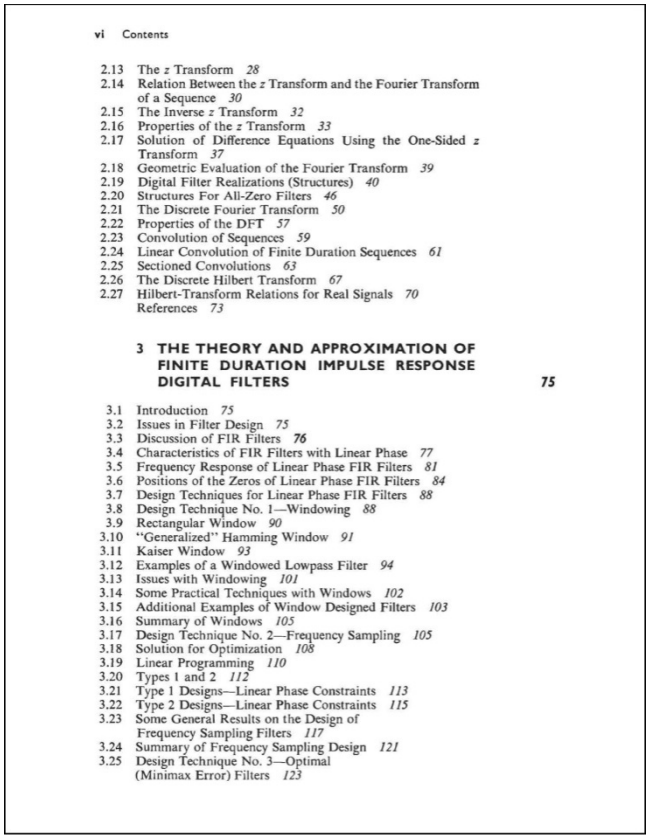
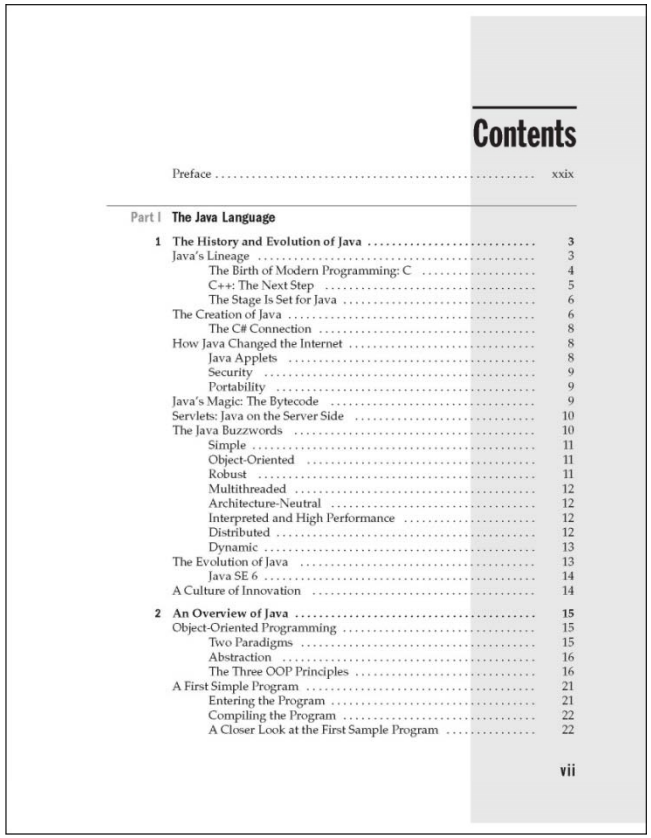


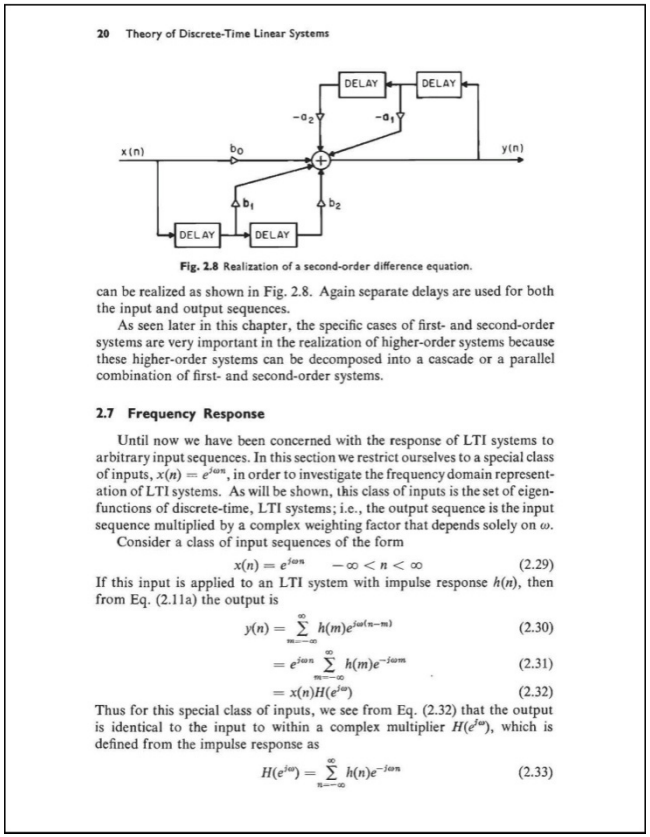
Table of Content page



Contents

Different types of pages

Text page-1



20 Theory of Discrete-Time Linear Systems

Fig. 2.8 Realization of a second-order difference equation.

can be realized as shown in Fig. 2.8. Again separate delays are used for both the input and output sequences.

As seen later in this chapter, the specific cases of first- and second-order systems are very important in the realization of higher-order systems because these higher-order systems can be decomposed into a cascade or a parallel combination of first- and second-order systems.

2.7 Frequency Response

Until now we have been concerned with the response of LTI systems to arbitrary input sequences. In this section we restrict ourselves to a special class of inputs, $x(n) = e^{j\omega n}$, in order to investigate the frequency domain representation of LTI systems. As will be shown, this class of inputs is the set of eigenfunctions of discrete-time, LTI systems; i.e., the output sequence is the input sequence multiplied by a complex weighting factor that depends solely on ω.

Consider a class of input sequences of the form

$$x(n) = e^{j\omega n} \qquad -\infty < n < \infty \tag{2.29}$$

If this input is applied to an LTI system with impulse response $h(n)$, then from Eq. (2.11a) the output is

$$y(n) = \sum_{m=-\infty}^{\infty} h(m)e^{j\omega(n-m)} \tag{2.30}$$

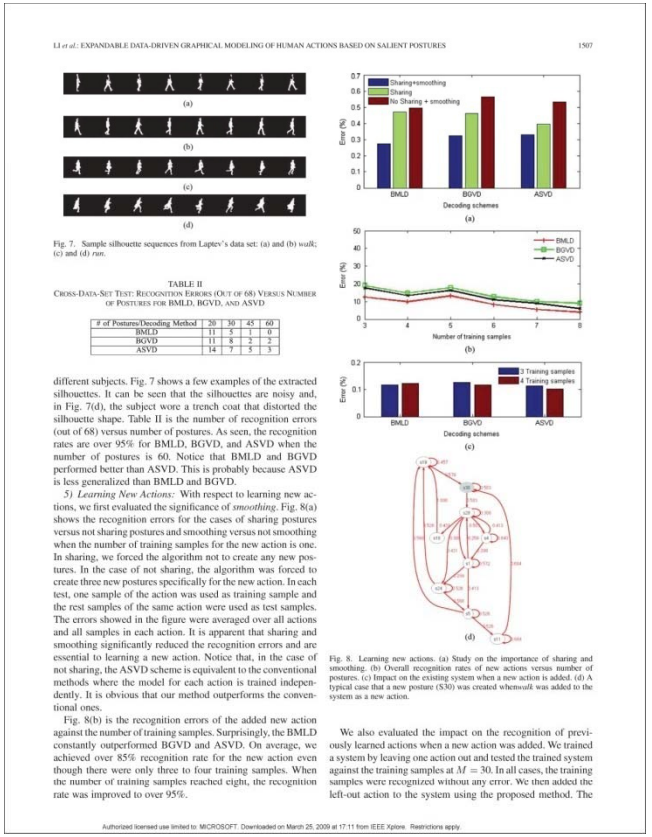
$$= e^{j\omega n} \sum_{m=-\infty}^{\infty} h(m)e^{-j\omega m} \tag{2.31}$$

$$= x(n)H(e^{j\omega}) \tag{2.32}$$

Thus for this special class of inputs, we see from Eq. (2.32) that the output is identical to the input to within a complex multiplier $H(e^{j\omega})$, which is defined from the impulse response as

$$H(e^{j\omega}) = \sum_{n=-\infty}^{\infty} h(n)e^{-j\omega n} \tag{2.33}$$

Text page-2



LI *et al.*: EXPANDABLE DATA-DRIVEN GRAPHICAL MODELING OF HUMAN ACTIONS BASED ON SALIENT POSTURES 1507

Fig. 7. Sample silhouette sequences from Laptev's data set: (a) and (b) *walk*; (c) and (d) *run*.

TABLE II
CROSS-DATA-SET TEST: RECOGNITION ERRORS (OUT OF 68) VERSUS NUMBER OF POSTURES FOR BMLD, BGVD, AND ASVD

# of Postures/Decoding Method	20	30	45	60
BMLD	11	5	1	0
BGVD	11	8	2	2
ASVD	14	7	5	3

different subjects. Fig. 7 shows a few examples of the extracted silhouettes. It can be seen that the silhouettes are noisy and, in Fig. 7(d), the subject wore a trench coat that distorted the silhouette shape. Table II is the number of recognition errors (out of 68) versus number of postures. As seen, the recognition rates are over 95% for BMLD, BGVD, and ASVD when the number of postures is 60. Notice that BMLD and BGVD performed better than ASVD. This is probably because ASVD is less generalized than BMLD and BGVD.

5) Learning New Actions: With respect to learning new actions, we first evaluated the significance of *smoothing*. Fig. 8(a) shows the recognition errors for the cases of sharing postures versus not sharing postures and smoothing versus not smoothing when the number of training samples for the new action is one. In sharing, we forced the algorithm not to create any new postures. In the case of not sharing, the algorithm was forced to create three new postures specifically for the new action. In each test, one sample of the action was used as training sample and the rest samples of the same action were used as test samples. The errors showed in the figure were averaged over all actions and all samples in each action. It is apparent that sharing and smoothing significantly reduced the recognition errors and are essential to learning a new action. Notice that, in the case of not sharing, the ASVD scheme is equivalent to the conventional methods where the model for each action is trained independently. It is obvious that our method outperforms the conventional ones.

Fig. 8(b) is the recognition errors of the added new action against the number of training samples. Surprisingly, the BMLD constantly outperformed BGVD and ASVD. On average, we achieved over 85% recognition rate for the new action even though there were only three to four training samples. When the number of training samples reached eight, the recognition rate was improved to over 95%.

Fig. 8. Learning new actions. (a) Study on the importance of sharing and smoothing. (b) Overall recognition rates of new actions versus number of postures. (c) Impact on the existing system when a new action is added. (d) A typical case that a new posture (S30) was created when *walk* was added to the system as a new action.

We also evaluated the impact on the recognition of previously learned actions when a new action was added. We trained a system by leaving one action out and tested the trained system against the training samples at $M = 30$. In all cases, the training samples were recognized without any error. We then added the left-out action to the system using the proposed method. The

Different types of pages

Text page-3

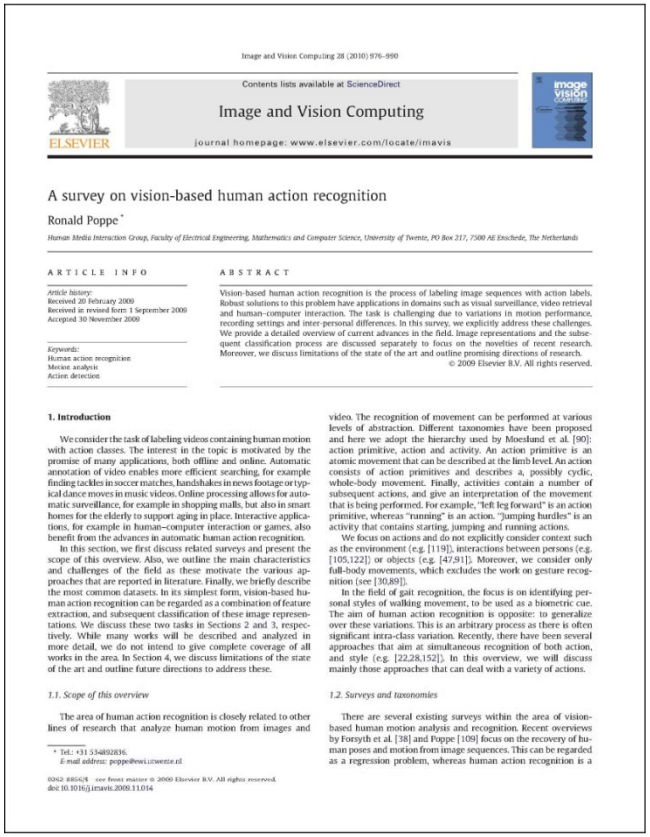


Image and Vision Computing 28 (2010) 976–990

Contents lists available at ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

A survey on vision-based human action recognition

Ronald Poppe *

Human Media Interaction Group, Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente, PO Box 217, 7500 AE Enschede, The Netherlands

ARTICLE INFO

Article history:
Received 20 February 2009
Received in revised form 1 September 2009
Accepted 30 November 2009

Keywords:
Human action recognition
Motion analysis
Action detection

ABSTRACT

Vision-based human action recognition is the process of labeling image sequences with action labels. Robust solutions to this problem have applications in domains such as visual surveillance, video retrieval and human–computer interaction. The task is challenging due to variations in motion performance, recording settings and inter-personal differences. In this survey, we explicitly address these challenges. We provide a detailed overview of current advances in the field. Image representations and the subsequent classification process are discussed separately to focus on the novelties of recent research. Moreover, we discuss limitations of the state of the art and outline promising directions of research.

1. Introduction

We consider the task of labeling videos containing human motion with action classes. The interest in the topic is motivated by the promise of many applications, both offline and online. Automatic annotation of video enables more efficient searching, for example finding tackles in soccer matches, handshakes in news footage or typical dance moves in music videos. Online processing allows for automatic surveillance, for example in shopping malls, but also in smart homes for the elderly to support aging in place. Interactive applications, for example in human–computer interaction or games, also benefit from the advances in automatic human action recognition.

In this section, we first discuss related surveys and present the scope of this overview. Also, we outline the main characteristics and challenges of the field as these motivate the various approaches that are reported in literature. Finally, we briefly describe the most common datasets. In its simplest form, vision-based human action recognition can be regarded as a combination of feature extraction, and subsequent classification of these image representations. We discuss these two tasks in Sections 2 and 3, respectively. While many works will be described and analyzed in more detail, we do not intend to give complete coverage of all works in the area. In Section 4, we discuss limitations of the state of the art and outline future directions to address these.

1.1. Scope of this overview

The area of human action recognition is closely related to other lines of research that analyze human motion from images and video. The recognition of movement can be performed at various levels of abstraction. Different taxonomies have been proposed and here we adopt the hierarchy used by Moeslund et al. [90]: action primitive, action and activity. An action primitive is an atomic movement that can be described at the limb level. An action consists of action primitives and describes a, possibly cyclic, whole-body movement. Finally, activities contain a number of subsequent actions, and give an interpretation of the movement that is being performed. For example, "left leg forward" is an action primitive, whereas "running" is an action. "Jumping hurdles" is an activity that contains starting, jumping and running actions.

We focus on actions and do not explicitly consider context such as the environment (e.g. [119]), interactions between persons (e.g. [105,122]) or objects (e.g. [47,91]). Moreover, we consider only full-body movements, which excludes the work on gesture recognition (see [30,89]).

In the field of gait recognition, the focus is on identifying personal styles of walking movement, to be used as a biometric cue. The aim of human action recognition is opposite: to generalize over these variations. This is an arbitrary process as there is often significant intra-class variation. Recently, there have been several approaches that aim at simultaneous recognition of both action, and style (e.g. [22,28,152]). In this overview, we will discuss mainly those approaches that can deal with a variety of actions.

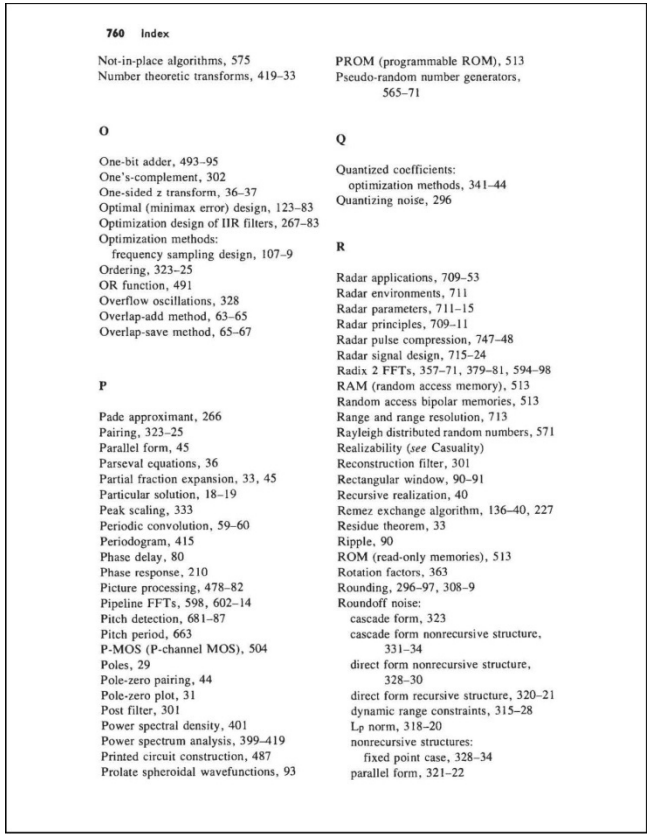
1.2. Surveys and taxonomies

There are several existing surveys within the area of vision-based human motion analysis and recognition. Recent overviews by Forsyth et al. [38] and Poppe [109] focus on the recovery of human poses and motion from image sequences. This can be regarded as a regression problem, whereas human action recognition is a

* Tel.: +31 534892836.
E-mail address: poppe@ewi.utwente.nl

0262-8856/$ – see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2009.11.014

Index page



Issues in document page scanning

- Resolution
- Back page impression
- Granular noise
- Blotted text *(specially in old documents)*
- Bending of pages at the binding
- Skew

  (due to placement of the page in the scanner)

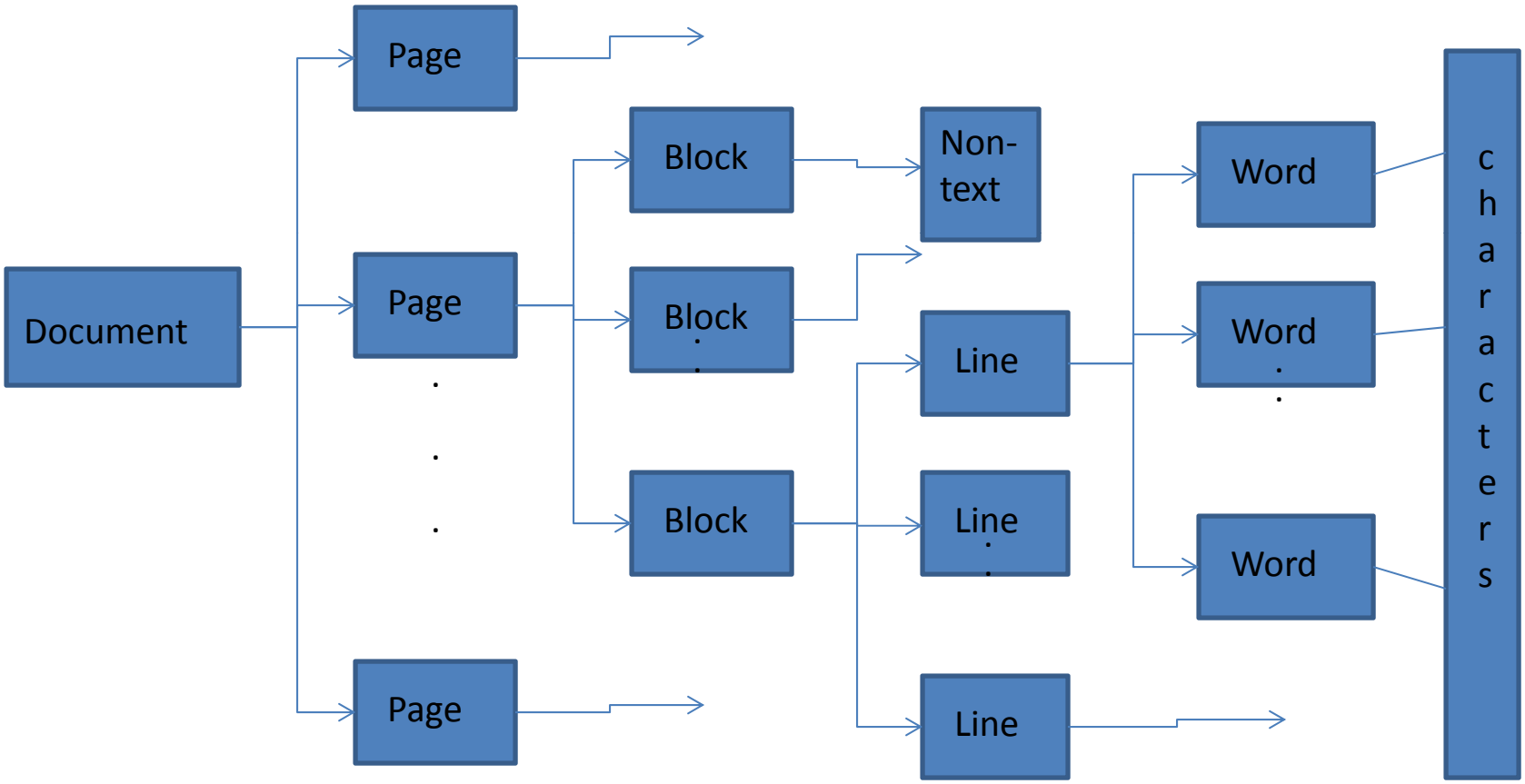
Entities of Document Page

- Text
  - Body text
    - Line → Word → Character
  - Heading
- Non-text
  - Half-tone
  - Table
  - Graphics or line drawing

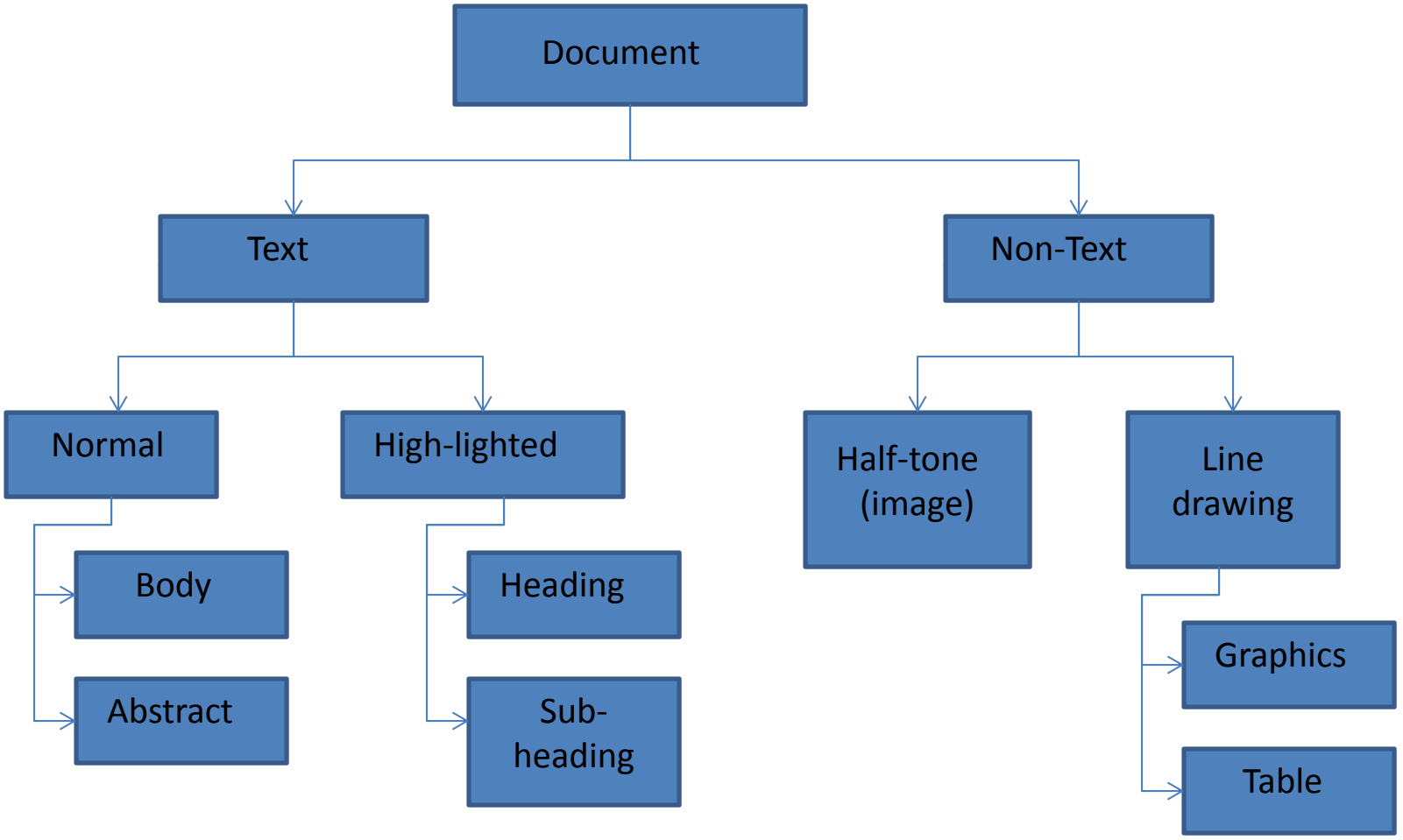
Entities of Document Page

- Each detected zone or block must be homogeneous in terms of content or entity
- Each zone will be input to one of the suitable modules based on entity.
  - OCR system
  - Image compressor
  - Vectorization system
- Output of these modules may be compiled and archived using suitable structure.

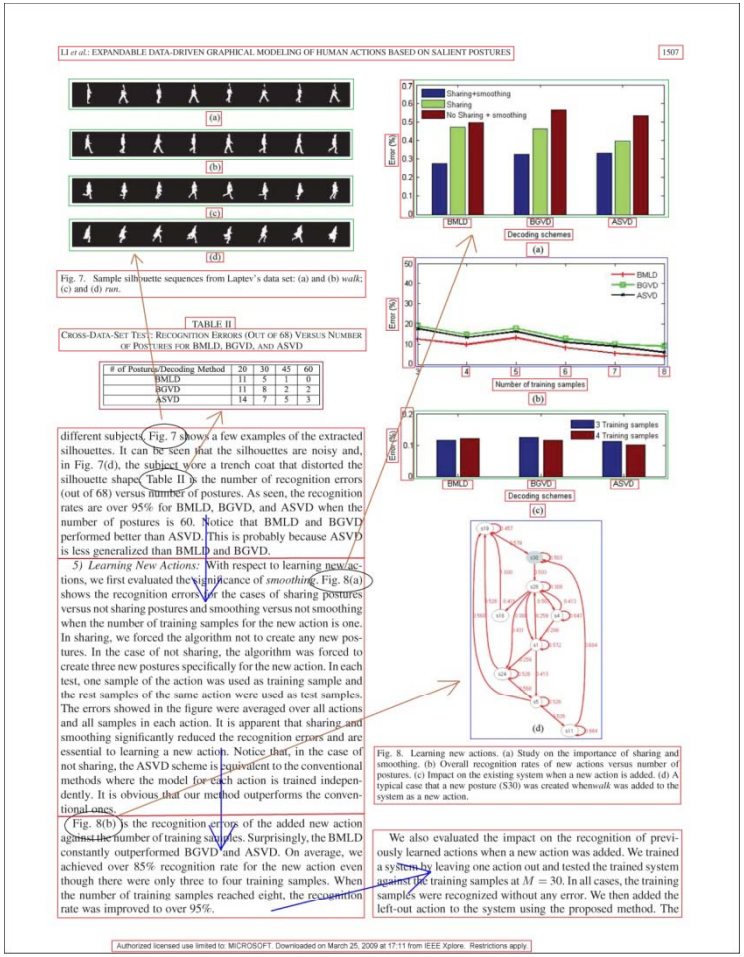
Geometrical / Physical structure



Logical structure



Logical structure



LI *et al.*: EXPANDABLE DATA-DRIVEN GRAPHICAL MODELING OF HUMAN ACTIONS BASED ON SALIENT POSTURES 1507

Fig. 7. Sample silhouette sequences from Laptev's data set: (a) and (b) *walk*; (c) and (d) *run*.

TABLE II
CROSS-DATA-SET TEST: RECOGNITION ERRORS (OUT OF 68) VERSUS NUMBER OF POSTURES FOR BMLD, BGVD, AND ASVD

# of Postures/Decoding Method	20	30	45	60
BMLD	11	5	1	0
BGVD	11	8	2	2
ASVD	14	7	5	3

different subjects. Fig. 7 shows a few examples of the extracted silhouettes. It can be seen that the silhouettes are noisy and, in Fig. 7(d), the subject wore a trench coat that distorted the silhouette shape. Table II is the number of recognition errors (out of 68) versus number of postures. As seen, the recognition rates are over 95% for BMLD, BGVD, and ASVD when the number of postures is 60. Notice that BMLD and BGVD performed better than ASVD. This is probably because ASVD is less generalized than BMLD and BGVD.

5) Learning New Actions: With respect to learning new actions, we first evaluated the significance of *smoothing*. Fig. 8(a) shows the recognition errors for the cases of sharing postures versus not sharing postures and smoothing versus not smoothing when the number of training samples for the new action is one. In sharing, we forced the algorithm not to create any new postures. In the case of not sharing, the algorithm was forced to create three new postures specifically for the new action. In each test, one sample of the action was used as training sample and the rest samples of the same action were used as test samples. The errors showed in the figure were averaged over all actions and all samples in each action. It is apparent that sharing and smoothing significantly reduced the recognition errors and are essential to learning a new action. Notice that, in the case of not sharing, the ASVD scheme is equivalent to the conventional methods where the model for each action is trained independently. It is obvious that our method outperforms the conventional ones.

Fig. 8(b) is the recognition errors of the added new action against the number of training samples. Surprisingly, the BMLD constantly outperformed BGVD and ASVD. On average, we achieved over 85% recognition rate for the new action even though there were only three to four training samples. When the number of training samples reached eight, the recognition rate was improved to over 95%.

Fig. 8. Learning new actions. (a) Study on the importance of sharing and smoothing. (b) Overall recognition rates of new actions versus number of postures. (c) Impact on the existing system when a new action is added. (d) A typical case that a new posture (S30) was created when *walk* was added to the system as a new action.

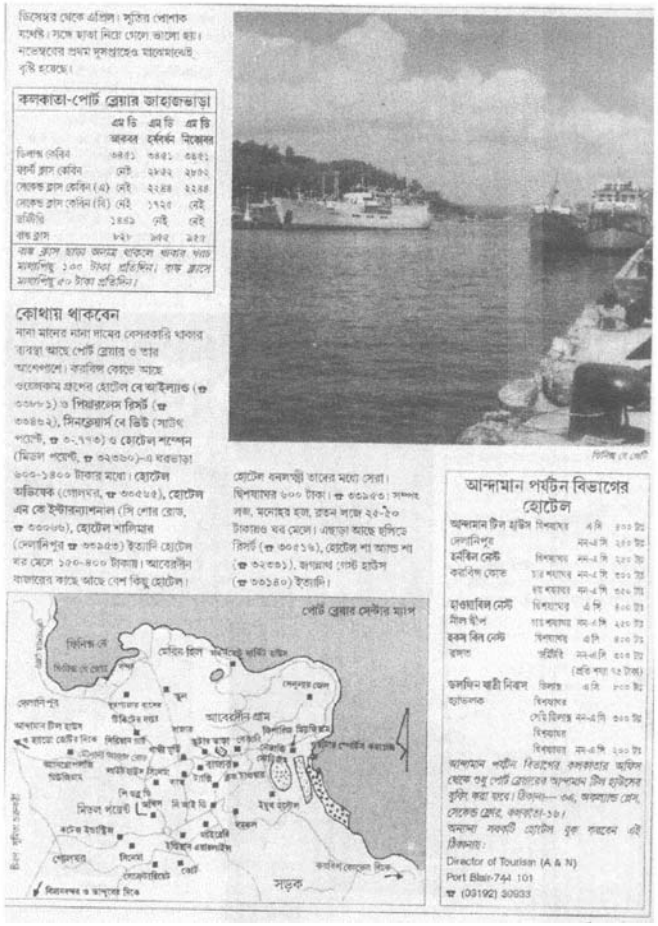
We also evaluated the impact on the recognition of previously learned actions when a new action was added. We trained a system by leaving one action out and tested the trained system against the training samples at $M = 30$. In all cases, the training samples were recognized without any error. We then added the left-out action to the system using the proposed method. The

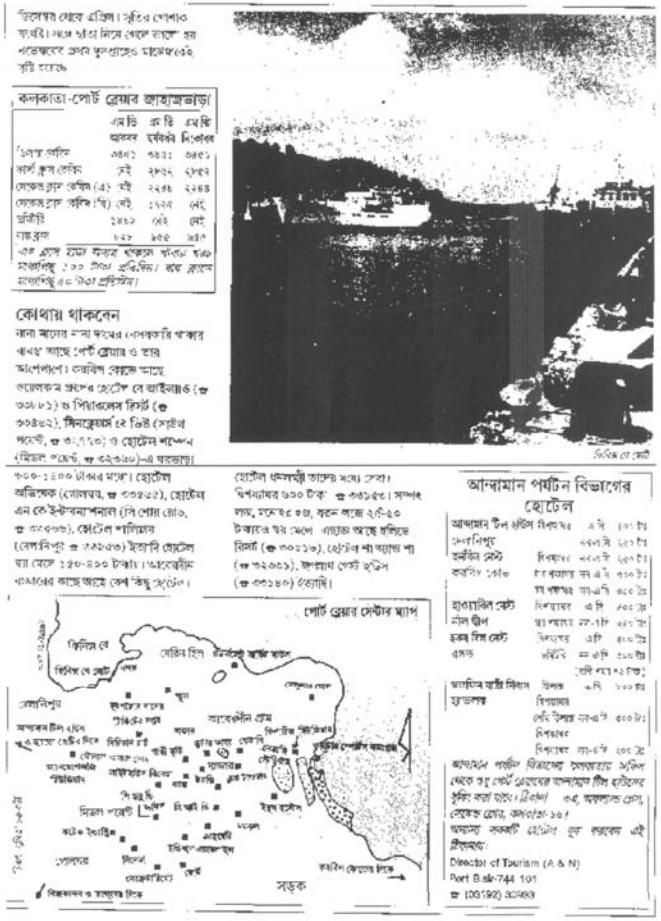
- Different entities:
  - Text (red box)
  - Halftone (green box)
  - Table (magenta box)
  - Line drawing (blue box)
- Reading direction
  (dark blue arrow)
- Link between entities
  (brown arrow)

Zone / block detection

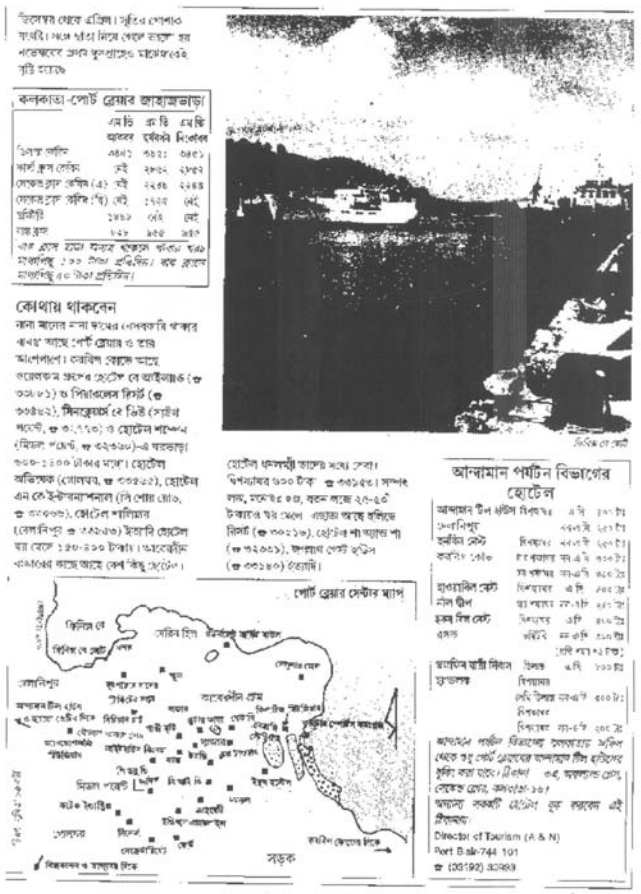
- One of the simple way is ***Projection method.***
- Algorithm
  - Take horizontal (or vertical) projection of foreground pixels. (may be implemented as pixel count)
  - If there exists a characteristic change in projection profile, put a horizontal (resp. vertical) separator.
  - Take horizontal and vertical direction alternately.
  - Continue, until above condition is satisfied.
- Works well for ***structured document***, usually the pages of technical journals, books, etc.

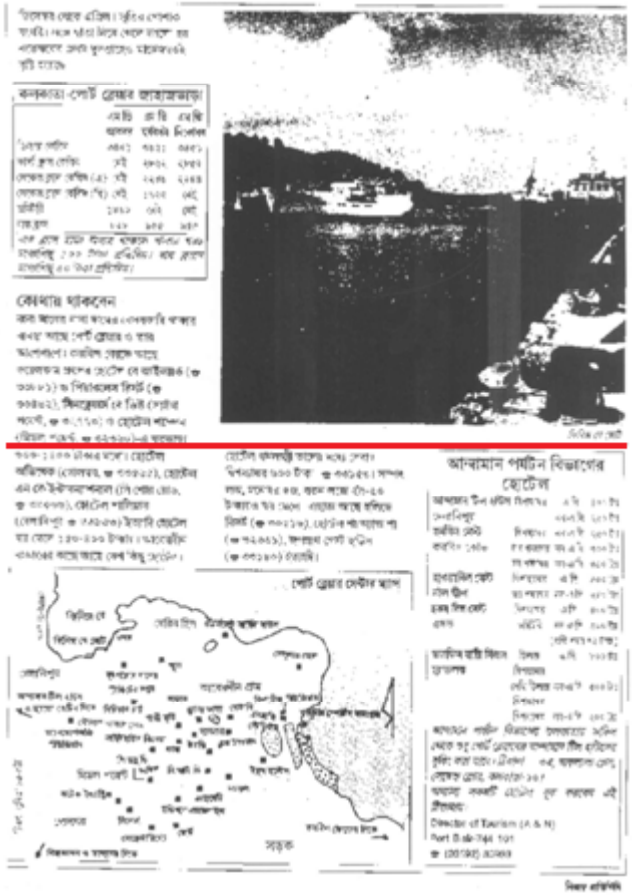
Projection Method: An Example



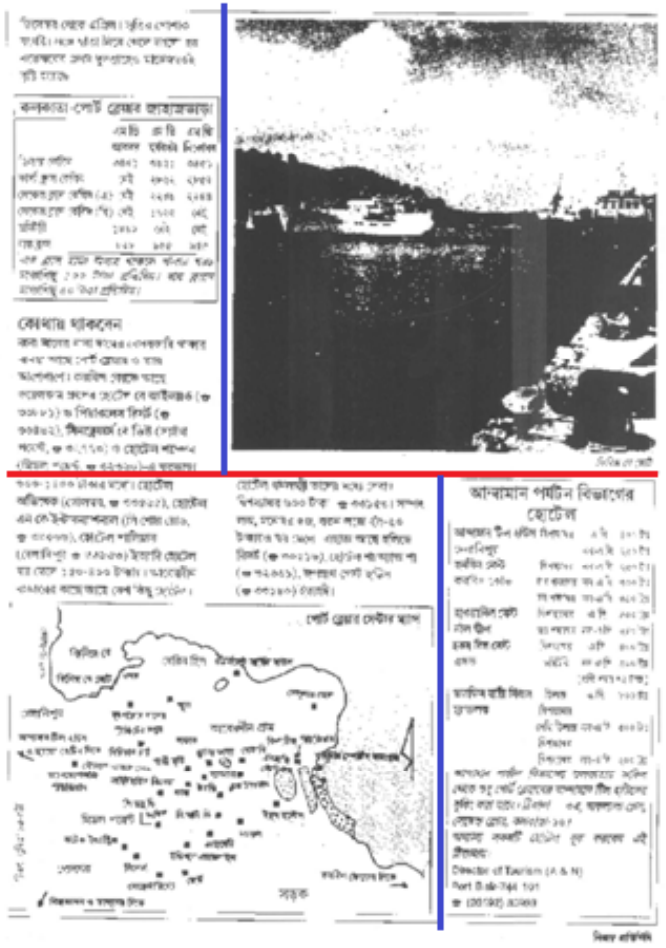


Example (contd.)

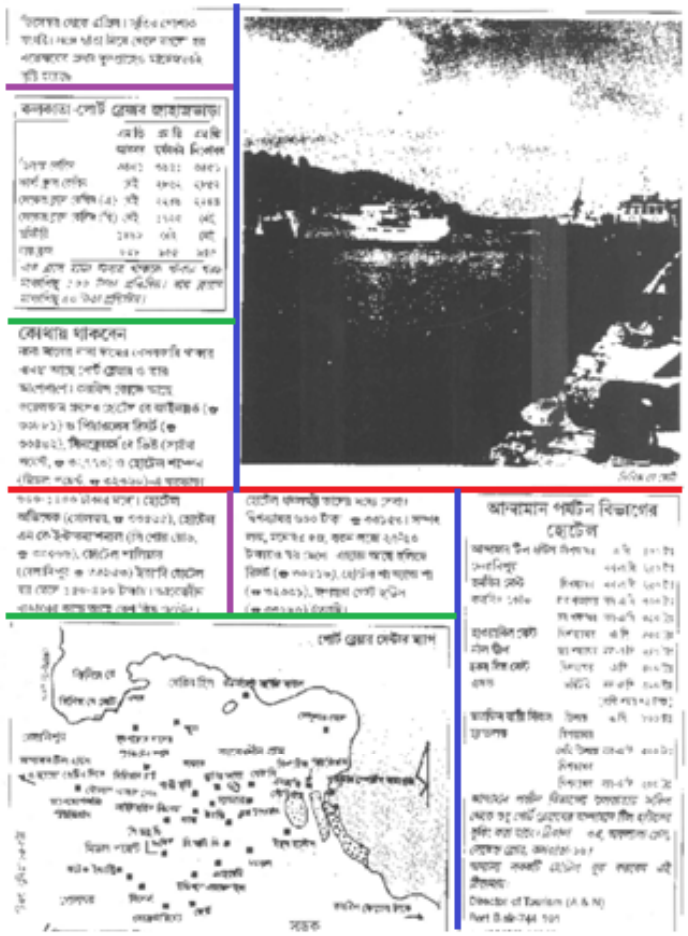


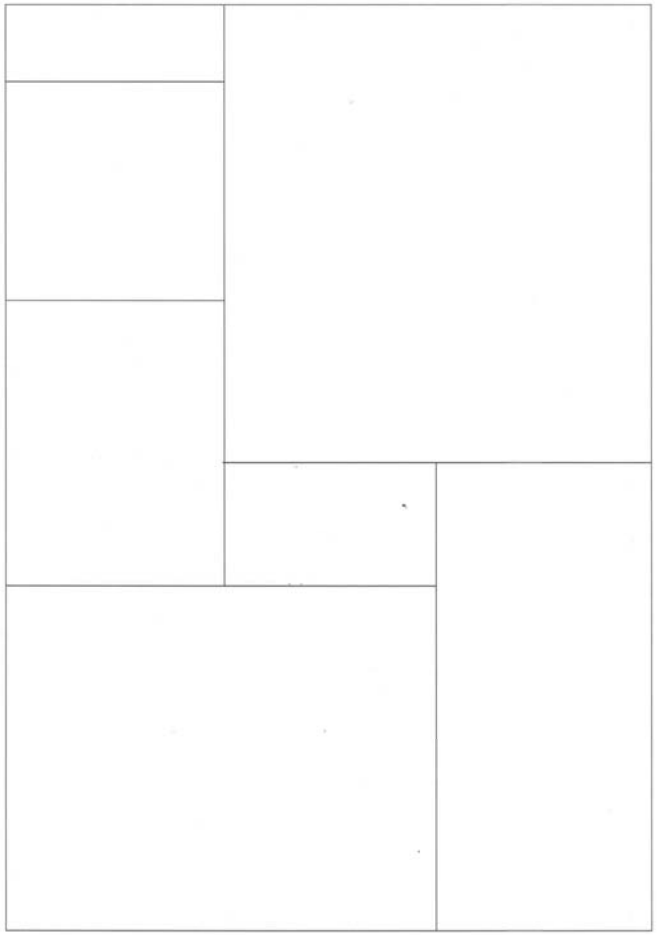


Example (contd.)



Example (contd.)





Problems of Projection method

- Cannot say what each block contains until further analysis.
  - Extract ***features*** from a zone
  - Recognize the zone content using a ***classifier***
- Results are highly dependent even on small skew in the scanned page.

Zone content recognition

Features:

- Black pixel ratio (no. of black pixel / zone area)
- Horizontal transition (black to white) count
- Vertical transition (black to white) count
- Normalized mean length of horizontal black pixel run
- Normalized mean length of vertical black pixel run
- Connected component ratio

Classifier:

- Two-class (text and non-text)

  SVM with RBF kernel (accuracy 94.89%)

Duong, Emptoz, Côté: Features for Printed Document Image Analysis, ICPR 2002.

Zone content recognition

- Functional classification of text blocks
  - Title / Heading, Sub-heading, Body text …
- Features:
  - complexity (measured by entropy)
  - visibility values (or relative boldness)
  - directional compactness (horizontal and vertical)
  - geometric characteristics (block height, width, etc.)
- Classifier:
  - K-means clustering followed by min. distance classifier

Bres, Eglin, and Gafneux, Unsupervised Clustering of Text Entities in Heterogeneous Grey Level Documents, ICPR, 2002.

Problems of Projection method

- Cannot say what each block contains until further analysis
  - Extract ***features*** from a zone
  - Recognize the zone content using a ***classifier***
- Results are highly dependent even on small skew in the scanned page
  - Detecting base line of each text line of the document
  - Determining orientation (slope) angle of base line
  - Estimation overall skew of the document page

Processing Tool

- Spatial domain operator that can handle shape information directly
- Mathematically well defined
- Neighborhood operator such that hardware implementation should be simple

Mathematical Morphology

- Mathematical morphological operators are good choice.

Objects

- All characters, figures, drawing, i.e., black components against white background

Structuring element

- Regular geometric figures:
  - mostly *line segment, square, circle,* etc.

Morphological Operations

Set theoretic operations (including union, intersection, etc.):

1. **Dilation**
2. **Erosion**
3. **Opening**
4. **Closing**

Morphological operator: Dilation

- Expands the objects.

$$A \oplus B = \{a + b \mid a \in A, b \in B\}$$

where A is an object and

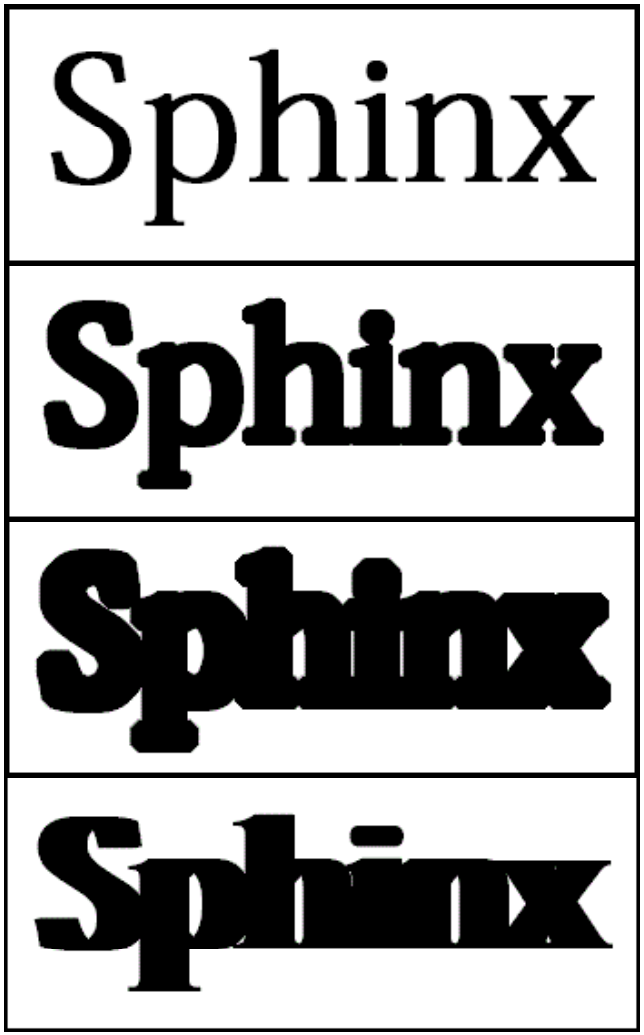
B is SE.

- **Properties:**

  Commutative,

  associative,

  distributive (over union), increasing



Morphological operator: Erosion

- Shrinks the objects.

$$A \Theta B = \{ p \mid B + p \in A \}$$

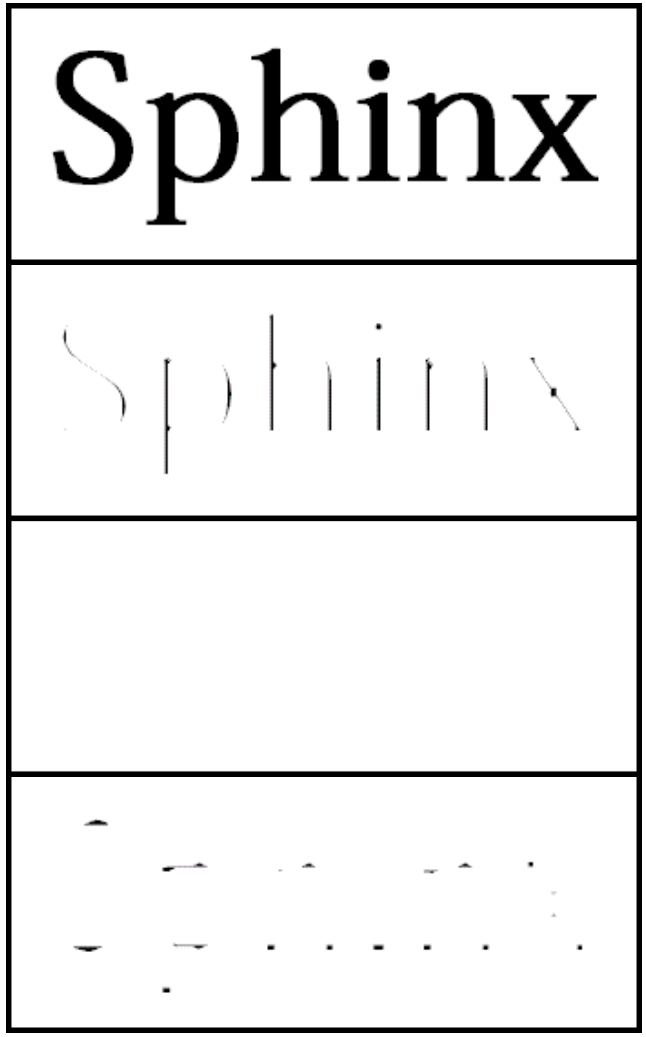
where A is an object and

B is SE.

- **Properties:**

Distributive (over intersection), increasing.

- *Dilation and erosion are dual.*



Morphological operator: Opening

- Removes objects or parts of it that cannot fit in SE.

$$A \circ B = (A \Theta B) \oplus B$$

where A is an object and

B is SE.

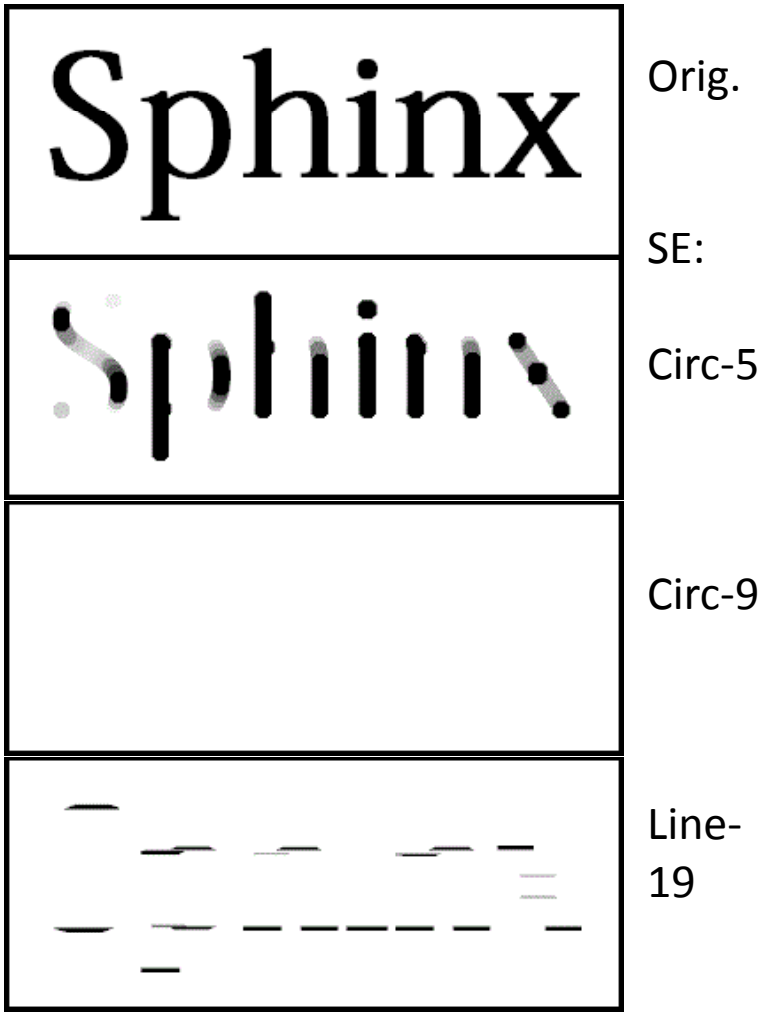
- **Properties:**

  Increasing,

  idempotent,

  anti-extensive.

- *It is a filter.*



Morphological operator: Closing

- Appends to objects parts of background if SE does not fit.

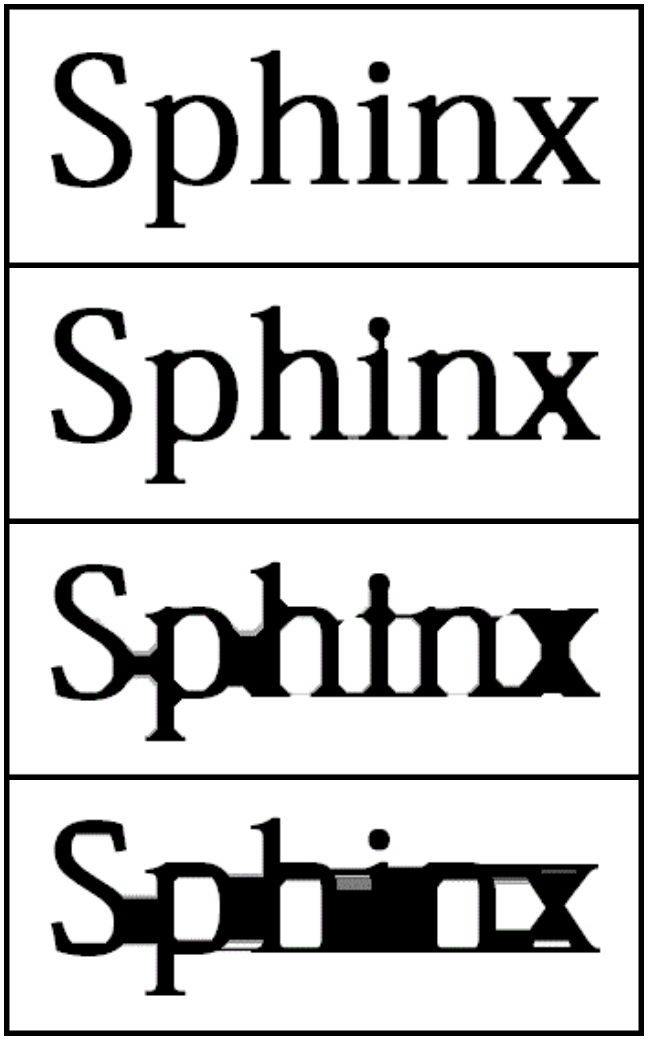
$$A \bullet B = \left(A \oplus B\right) \Theta B$$

where A is an object and

B is SE.

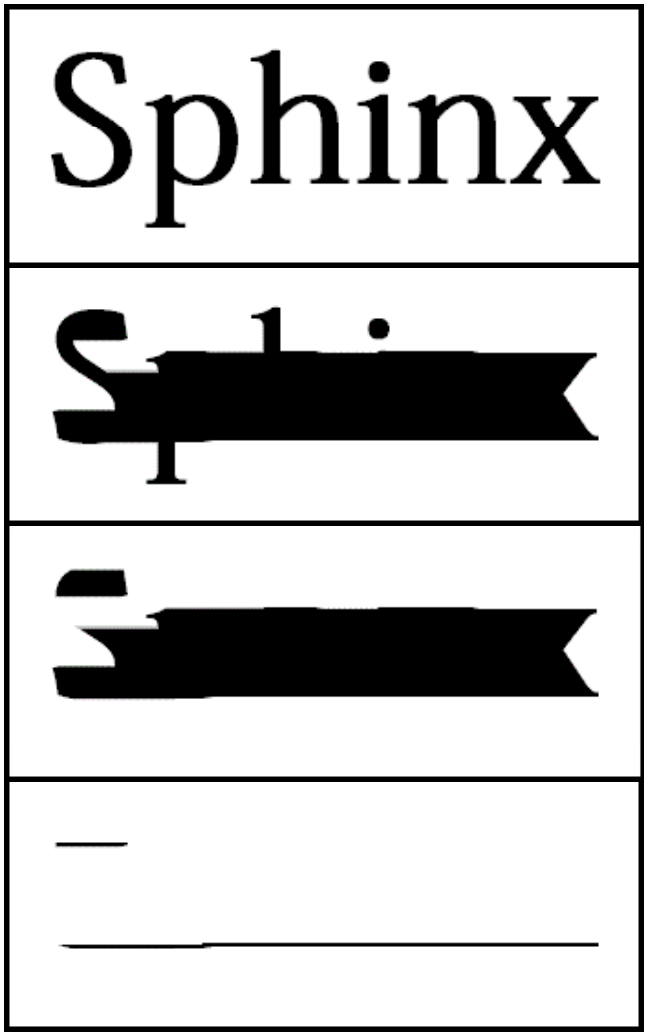
- **Properties:**

  Increasing, idempotent, and extensive.
- It is a filter.
- *Opening & closing are dual.*



Detecting base line

- Close the original image with line SE of suitable length.
- Open the close image with same line SE.
- Detect black to white transition in vertical scan.



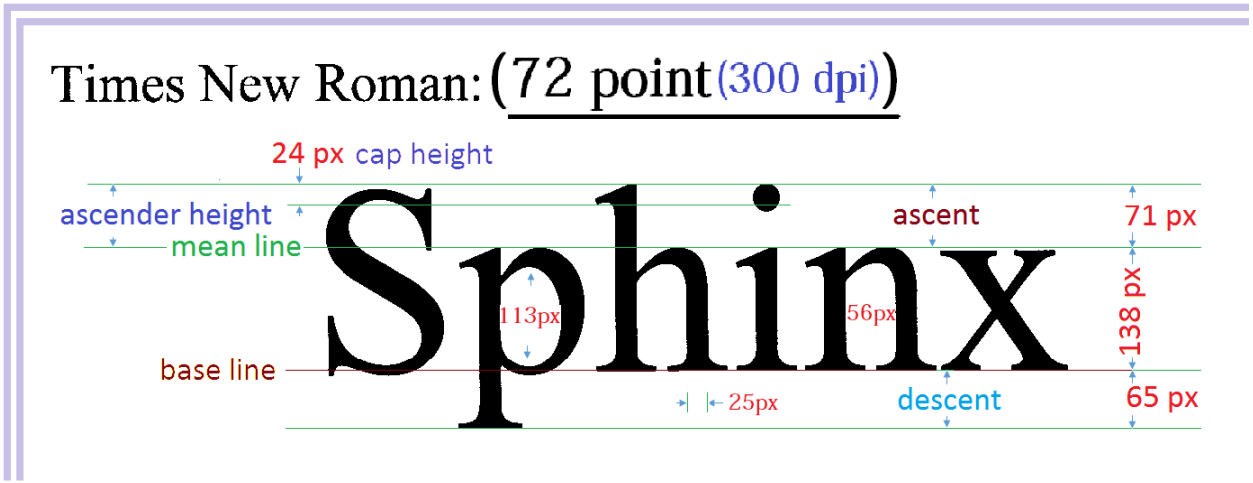
Font

- Traditionally, in metal typesetting, a **font** is a particular ***size***, ***weight*** and ***style*** of a typeface.
- The weight of a particular font is the thickness of the character outlines relative to their height.
- Font size is measured in ***point*** unit.

1 point in	*is equal to ...*
typographic units	1/12 picas
imperial/US units	1/72 inch
metric (SI) units	0.3528 mm

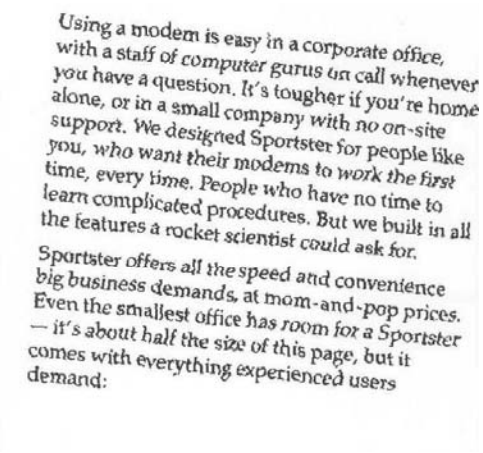
Size related parameters

- X-height or ***corpus*** height
- Ascender
- Descender



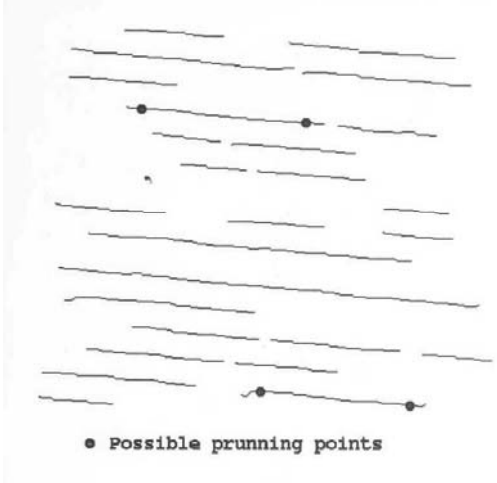
- Scan resolution (in dpi)
- Font style: bold, italics, ornamental

Skew correction: An example



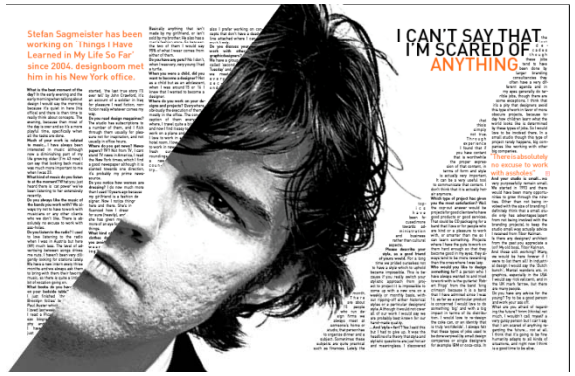
Using a modem is easy in a corporate office, with a staff of computer gurus on call whenever you have a question. It's tougher if you're home alone, or in a small company with no on-site support. We designed Sportster for people like you, who want their modems to work the first time, every time. People who have no time to learn complicated procedures. But we built in all the features a rocket scientist could ask for.

Sportster offers all the speed and convenience big business demands, at mom-and-pop prices. Even the smallest office has room for a Sportster — it's about half the size of this page, but it comes with everything experienced users demand:



• Possible prunning points

Pages with complex layout



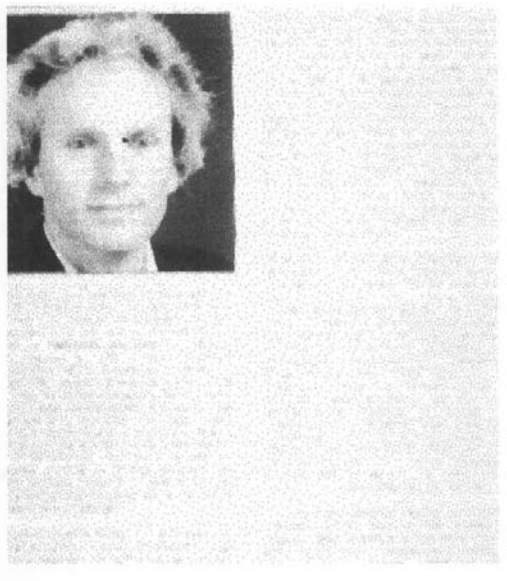
Morphological algorithm

- Text region is composed of small objects (characters) placed in regular interval.
- Opening the image with small SE removes the thin object parts (strokes of character), but has insignificant effect on large objects in half-tone etc.
- Closing the image with small SE fills in white holes in small objects (space within and between character), but has insignificant effect on large white space or half-tone.
- Thus difference between closed and opened image highlights the text region.
- Difference image is thresholded to detect text region.

Morphological approach: An example

(a) Original image

(b) Closed image

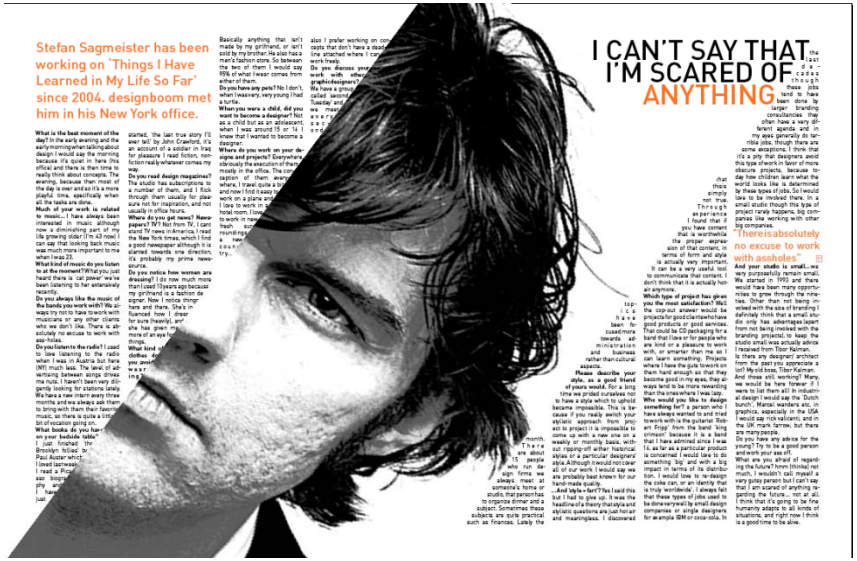


(c) Opened image

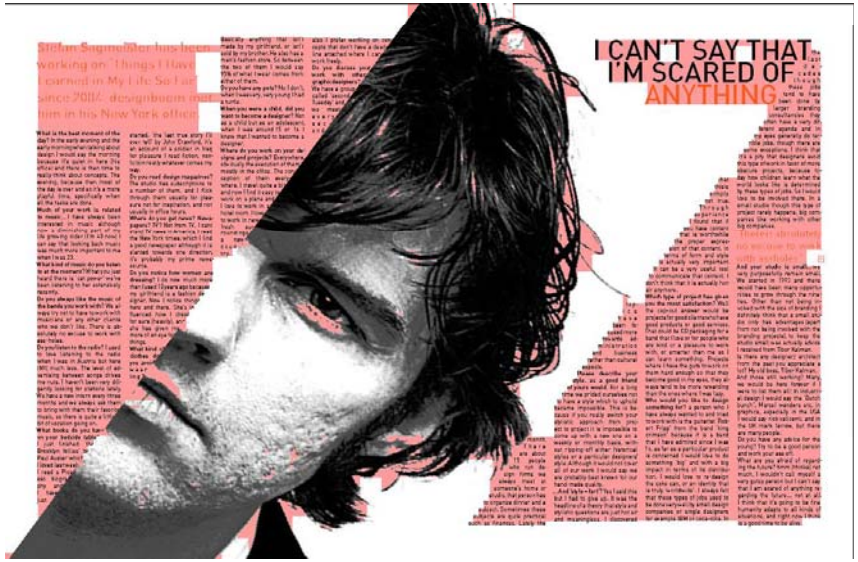
QUESTION: Size of structuring element?

Results

Input test image



Resultant (labeled) image



Results

Input test image

Resultant (labeled) image

Results

Input test image

Resultant (labeled) image

Results

Input test image

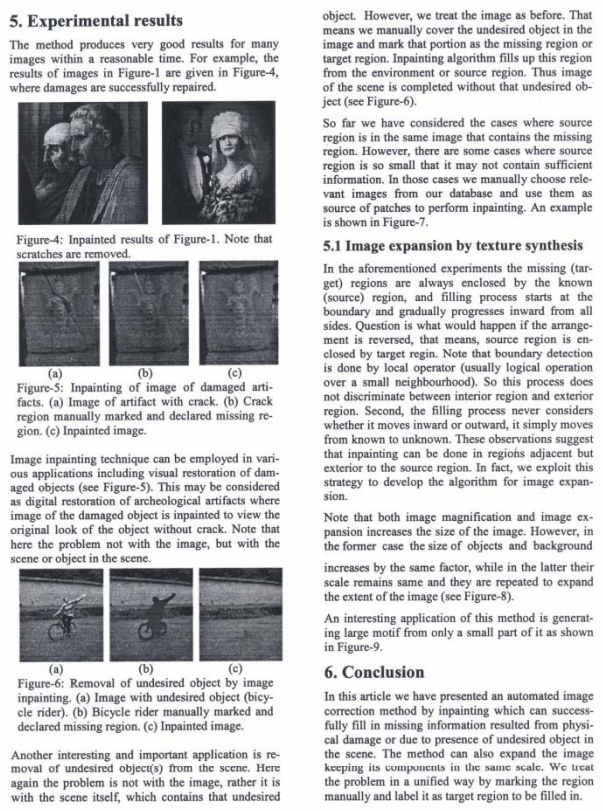
Resultant (labeled) image

Deep learning

- Popular technique for unsupervised feature extraction for supervised applications
  - Ex. object recognition.
- Utilizes HUGE number of instances to train relatively simpler system to perform more complicated task.
- Training samples may be outcome of controlled or uncontrolled data acquisition.
- Requires very high computational resources for implementing a reasonably meaningful system.

Detect text area using CNN

Input: *A document image*



5. Experimental results

The method produces very good results for many images within a reasonable time. For example, the results of images in Figure-1 are given in Figure-4, where damages are successfully repaired.

Figure-4: Inpainted results of Figure-1. Note that scratches are removed.

(a) (b) (c)

Figure-5: Inpainting of image of damaged artifacts. (a) Image of artifact with crack. (b) Crack region manually marked and declared missing region. (c) Inpainted image.

Image inpainting technique can be employed in various applications including visual restoration of damaged objects (see Figure-5). This may be considered as digital restoration of archeological artifacts where image of the damaged object is inpainted to view the original look of the object without crack. Note that here the problem not with the image, but with the scene or object in the scene.

(a) (b) (c)

Figure-6: Removal of undesired object by image inpainting. (a) Image with undesired object (bicycle rider). (b) Bicycle rider manually marked and declared missing region. (c) Inpainted image.

Another interesting and important application is removal of undesired object(s) from the scene. Here again the problem is not with the image, rather it is with the scene itself, which contains that undesired object. However, we treat the image as before. That means we manually cover the undesired object in the image and mark that portion as the missing region or target region. Inpainting algorithm fills up this region from the environment or source region. Thus image of the scene is completed without that undesired object (see Figure-6).

So far we have considered the cases where source region is in the same image that contains the missing region. However, there are some cases where source region is so small that it may not contain sufficient information. In those cases we manually choose relevant images from our database and use them as source of patches to perform inpainting. An example is shown in Figure-7.

5.1 Image expansion by texture synthesis

In the aforementioned experiments the missing (target) regions are always enclosed by the known (source) region, and filling process starts at the boundary and gradually progresses inward from all sides. Question is what would happen if the arrangement is reversed, that means, source region is enclosed by target regin. Note that boundary detection is done by local operator (usually logical operation over a small neighbourhood). So this process does not discriminate between interior region and exterior region. Second, the filling process never considers whether it moves inward or outward, it simply moves from known to unknown. These observations suggest that inpainting can be done in regions adjacent but exterior to the source region. In fact, we exploit this strategy to develop the algorithm for image expansion.

Note that both image magnification and image expansion increases the size of the image. However, in the former case the size of objects and background increases by the same factor, while in the latter their scale remains same and they are repeated to expand the extent of the image (see Figure-8).

An interesting application of this method is generating large motif from only a small part of it as shown in Figure-9.

6. Conclusion

In this article we have presented an automated image correction method by inpainting which can successfully fill in missing information resulted from physical damage or due to presence of undesired object in the scene. The method can also expand the image keeping its components in the same scale. We treat the problem in a unified way by marking the region manually and label it as target region to be filled in.

Output:*Text / Non-text area*

Solution strategy

Transforming the problem into a classification Problem.

- Divide the Input image into MxM patches.
- **Input:** Image patch of size MxM
- **Output:** Text, Non text, and Ambiguous
  - Text: if >80% of the patch has text
  - Non-text: if <20% of the patch has text area
  - Ambiguous: otherwise

Training data

Cytochrome P450 3A Enzymes Catalyze the O^6-Demethylation of Thebaine, a Key Step in Endogenous Mammalian Morphine Biosynthesis

Valerie M. Kramlinger, Mónica Alvarado Rojas, Tatsuyuki Kanamori, and F. Peter Guengerich

From the Department of Biochemistry, School of Medicine, Vanderbilt University School of Medicine, Nashville, Tennessee 37232-0146

Running title: *P450 3A and morphine synthesis*

To whom correspondence should be addressed: Prof. F. Peter Guengerich, Department of Biochemistry, Vanderbilt University School of Medicine, 638B Robinson Research Building, 2200 Pierce Avenue, Nashville, Tennessee 37232-0146, Telephone: (615) 322-2261; FAX: (615) 343-0704; E-mail: f.guengerich@vanderbilt.edu

Keywords: Cytochrome P450, biosynthesis, natural product, oxygenase, morphine, alkaloids

ABSTRACT

Morphine, first characterized in opium from the poppy *Papaver somniferum*, is one of the strongest known analgesics. Endogenous morphine has been identified in several mammalian cells and tissues. The synthetic pathway of morphine in the opium poppy has been elucidated, and the presence of common intermediates in plants and mammals suggests that biosynthesis occurs through similar pathways (beginning with the amino acid L-tyrosine), and the pathway has been completely delineated in plants. Some of the enzymes in the mammalian pathway have been identified and characterized. Two of the latter steps in the morphine biosynthesis pathway are demethylation of thebaine at the O^3- and the O^6-positions, the latter of which has been difficult to demonstrate. The plant enzymes responsible for both the O^3- and the O^6-demethylation are members of the Fe^{II}/α-ketoglutarate-dependent dioxygenase family. Previous studies showed that human cytochrome P450 (P450) 2D6 can catalyze thebaine O^3-demethylation. We report that demethylation of thebaine at the O^6-position is selectively catalyzed by human P450s 3A4 and 3A5, with the latter being more efficient, and rat P450 3A2. Our results do not support O^6-demethylation of thebaine by an Fe^{II}/α-ketoglutarate-dependent dioxygenase. In rat brain microsomes, O^6-demethylation was inhibited by ketoconazole, but not sulfaphenazole, suggesting that P450 3A enzymes are responsible for this activity in the brain. An alternate pathway to morphine, oripavine O^6-demethylation, was not detected. The major enzymatic steps in mammalian morphine synthesis have now been identified.

Morphine is the principal benzylisoquinoline alkaloid produced in the opium poppy, *Papaver somniferum*. Its mechanism of action is as an agonist for the μ-opioid receptors that are distributed throughout the brain. Activation of these receptors is associated with analgesia, sedation, euphoria, physical dependence, and respiratory depression. The μ-binding sites are discretely distributed in the human brain, with high densities in the posterior amygdala, hypothalamus, thalamus, nucleus caudatus, putamen, and certain cortical areas (1).

A considerable body of evidence exists that morphine is present in the tissues of various animals that have not been medicated with morphine or other related opioids. In mammals, morphine has been detected in skin, lung, spinal cord, and, most notably, in the brain (2-4). The presence of morphine in brain is of particular interest due to the presence of the μ-opioid receptors. In the rat, morphine levels in structures of the brain have been quantified and range from 26 fmol/g tissue (found globally in the brain) to 7.2 pmol/g tissue measured in the hypothalamus (3,5-7). One study quantified morphine in the temporal lobe from one human brain tissue at 3.4 ng/g tissue (8). The presence of morphine in brain tissue in many mammals and the

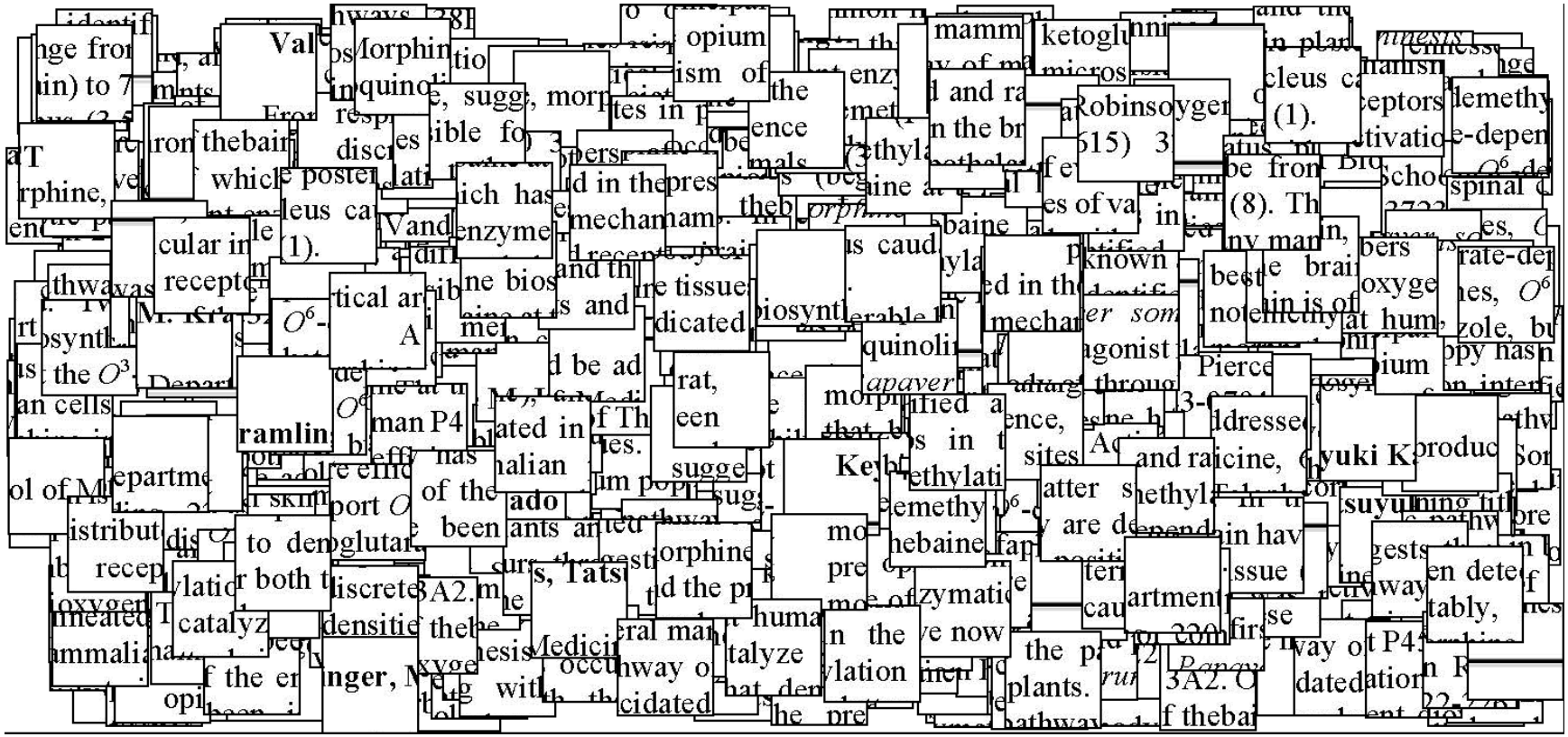
Prepare training data

INPUT: document images with manually labeled text area.

- From each image, overlapping patches of size 100x100 are taken (stride along x, y is 20) and resized to 50x50
- From each image, overlapping patches of size 50x50 are taken (stride along x, y is 10)
- Each 50x50 patch is divided into 4 patches of size 25x25 and are resized back to 50x50.
- We get total number of **825670** patches of size 50x50 as training data from 8 images.

Label: as described before.

Training blocks: Example



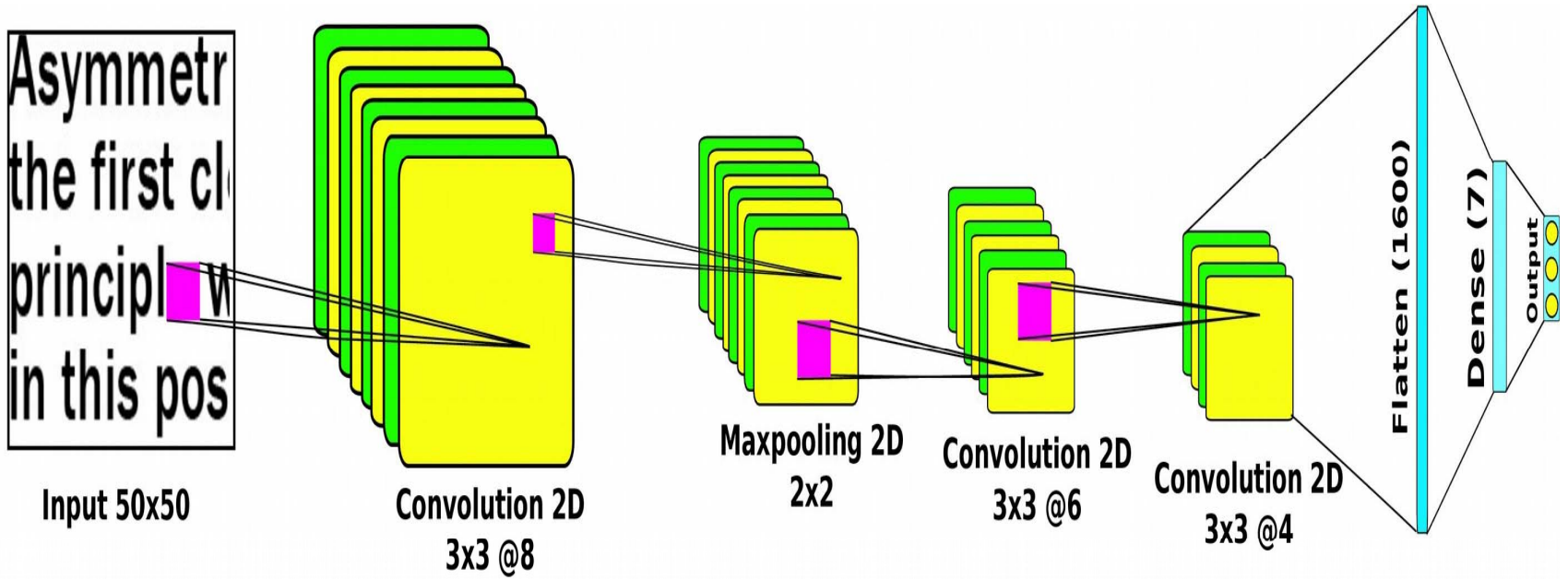
Model description

Input: 50x50 Patch of gray scale.

Layer (type)	Output Shape	Param #
Convolution2D(3x3 @8)	(8, 48, 48)	80
MaxPooling2D(2x2)	(8, 24, 24)	0
Convolution2D(3x3 @6)	(6, 22, 22)	438
Convolution2D(3x3 @4)	(4, 20, 20)	220
Flatten	(1600)	0
Dense(7)	(7)	11207
Activation(Sigmoid)	(7)	0
Dense(3)	(3)	24
Activation(Softmax)	(3)	0

Total parameters: 11969

Model description



Training the model

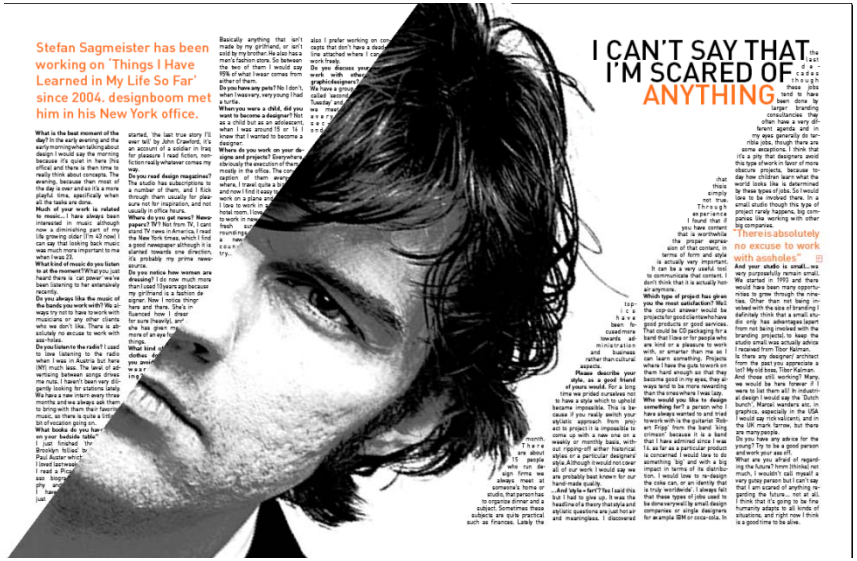
- Number of epoch: 200
- Batch size: 100
- Learning Rate: 0.01
- Learning weight decay: 0.95
- Optimizer: Stochastic gradient descent
- Loss function: Mean squared error

Testing

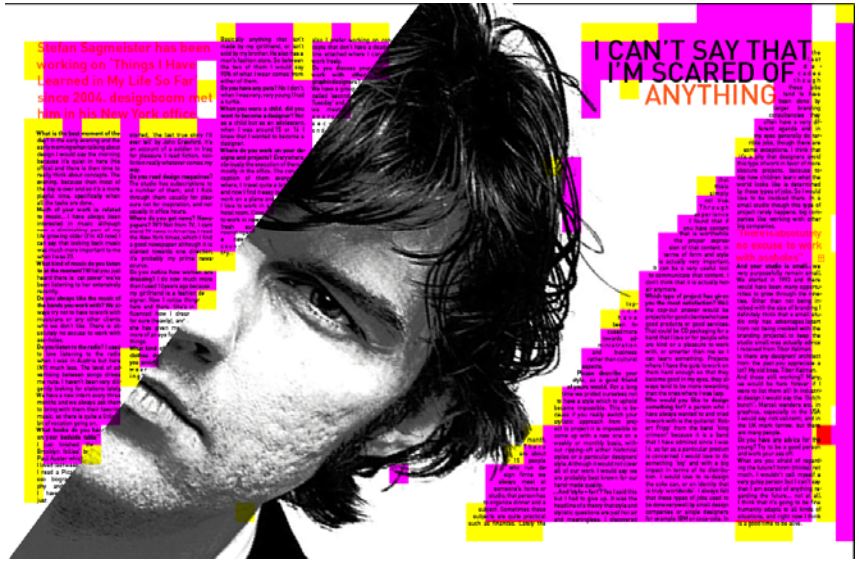
- Input: A test image
- Take 50x50 patch and submit it to the trained model
- If predicted class is text, color that patch as pink.
- If predicted class is non-text, color the patch as white.
- If predicted class is ambiguous, then
  - Divide that patch into 4 patches of size 25x25 and resize to 50x50 and submitted to the model.
  - If that 50x50 patch is again ambiguous, then color that patch as yellow (Ideally it should be done recursively until we get no ambiguous patch)
  - Else color the patch as according to text or non-text class.

Results

Input test image



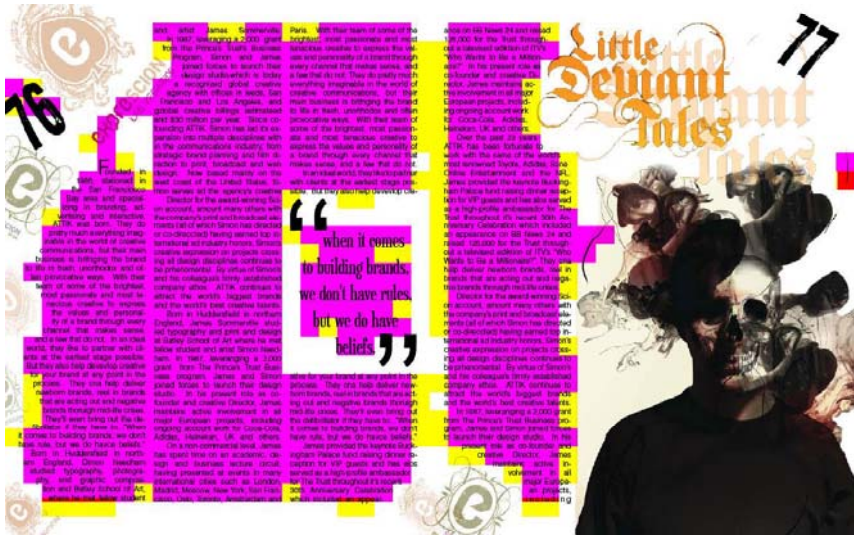
Resultant (labeled) image



Results

Input test image

Resultant (labeled) image



Results

Input test image

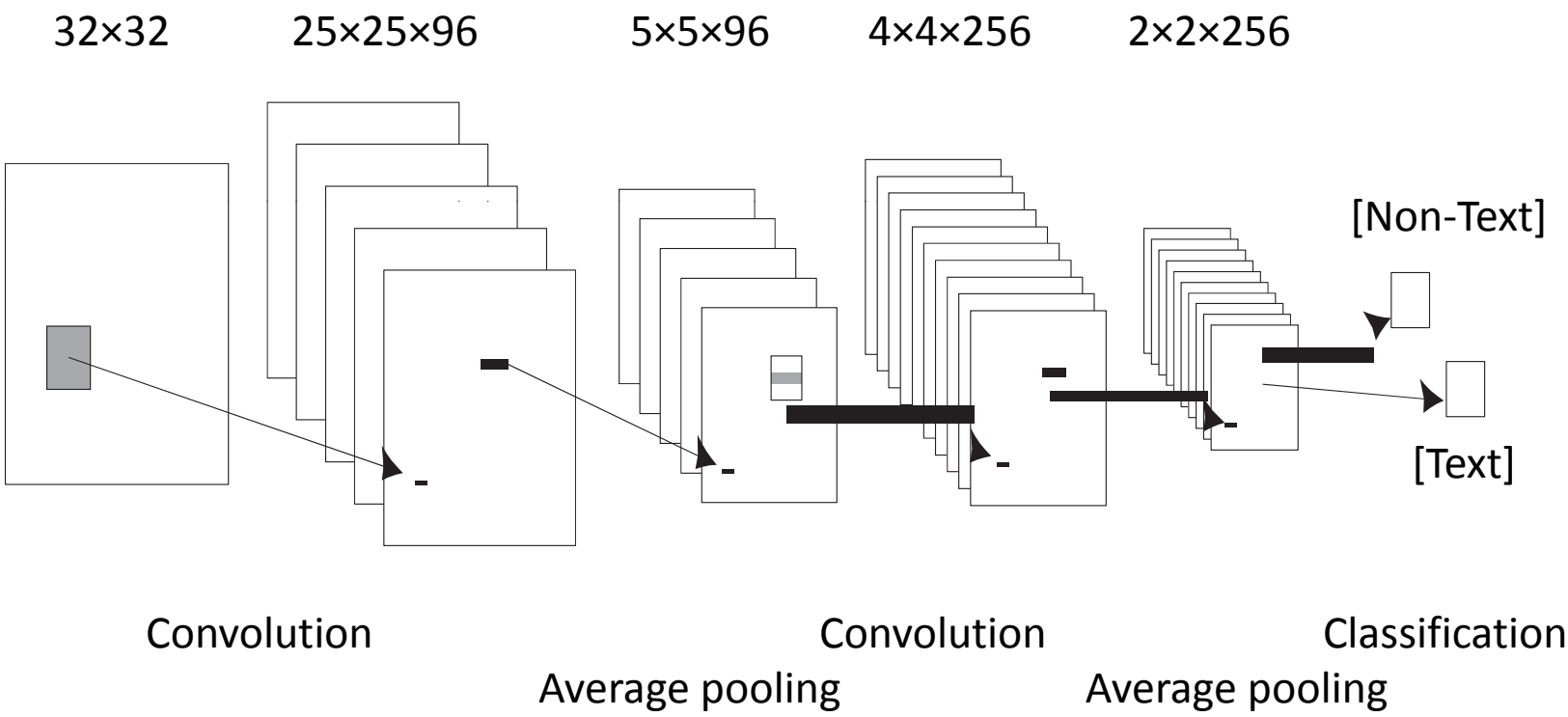
Resultant (labeled) image

Results

Input test image

Resultant (labeled) image

An improved network



Wang, Wu, Coates and Ng**, End-to-End Text Recognition with Convolutional Neural Networks**, *ICPR 2012.*

Comparative results

Simpler system

Wang et al.

Comparative results

Simpler system

Wang et al.

Benchmark database

- UW-I, II, III databeases, Developed at University of Washington, Seattle, USA in 1996.
- Widely used earliest database with1620 pages
- Zones contain text, non-text such as halftone, line drawing, math and chemical equation.
- The database also contains
  - *Page condition file :* skew angle, noise.
  - *Page attribute file :* dominant font and other content.
  - *Page bounding box file :* location and size of zones.

http://isis-data.science.uva.nl/events/dlia//datasets/uwash3.html

Benchmark database

- Mediateam document database
- Developed at University of Oulu, Finland in1998.
- One of the early databases containing

Pattern type	Samples
Text	4811
Graphics	735
Image	161
Composite	219

Duong, Emptoz, Côté: Features for Printed Document Image Analysis, ICPR 2002.

Benchmark database

- Pattern Recognition and Image Analysis (PRImA) Layout Analysis dataset
- Developed at University of Salford, Manchester
- 1240 ground-truthed pages from magazines (1085 pages) and technical journals (155 pages)
- Used in following contests
  - ICDAR 2015 Recognition of Documents with Complex Layouts (RDCL2015)
  - ICDAR2013 Historical Newspaper Layout Analysis (HNLA2013)
  - ICDAR2011 Historical Document Layout Analysis (HDLAC 2011)

http://www.primaresearch.org/datasets/Layout_Analysis

Benchmark database

- Historical Newspaper dataset (ENP dataset)
- Developed at University of Salford, Manchester in *Europeana Newspapers Project*
- 500 ground-truthed pages covering
  - 13 languages (German, french, English, Estonian, etc.)
  - 17^{th}, 18^{th}, 19^{th} and 20^{th} centuries
- Contains (total regions 61,619) including
  - 1,497 image zones
  - 208 table zones
  - 46,889 text zones

Clausner et. Al, The ENP Image and Ground Truth Dataset of historical newspaper, ICDAR 2015.

Performance evaluation

- A document page *D* may be represented as a *m* tuple.

$$D = (E_1, E_2, \ldots, E_m)$$

where E_i *s* are entities such as text, tables, half-tone, etc.

- Each entity has a unique property denoted by Prop.(E_i).
- Document page image domain *X* has *n* bounding boxes B_j *(j=1,…, n) with such that:*

$(i)\ \bigcup_1^n B_j \subseteq X$

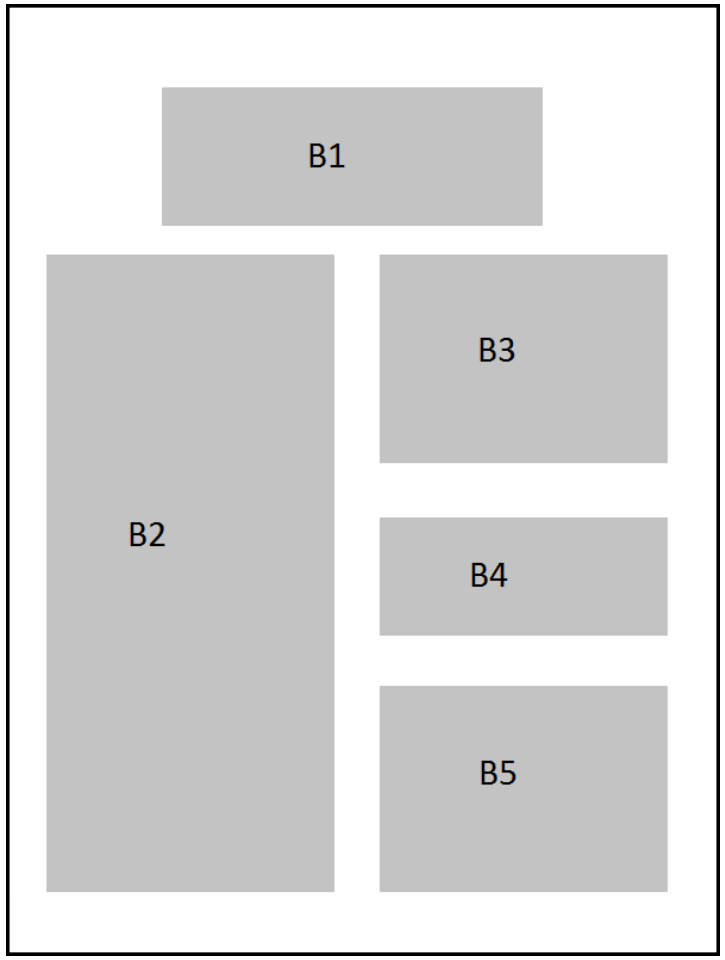
$(ii)\ B_j \cap B_k = \phi \ \text{ for } \ j \neq k$

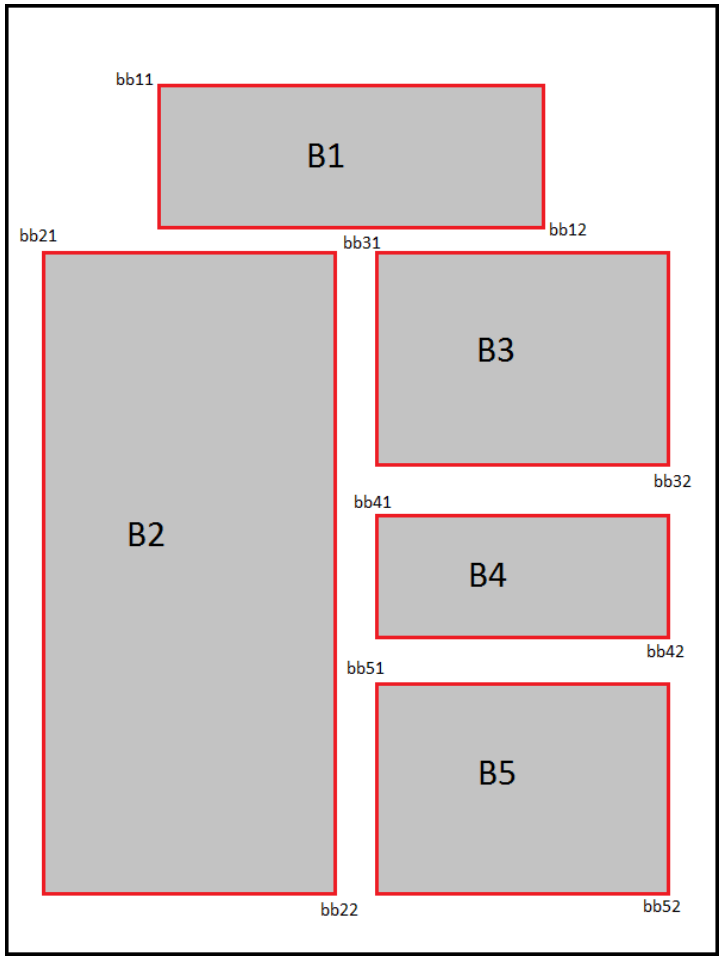
(iii) For every j there exists one and only one i such that

$\text{Prop.}(B_j) = \text{Prop.}(E_i)$

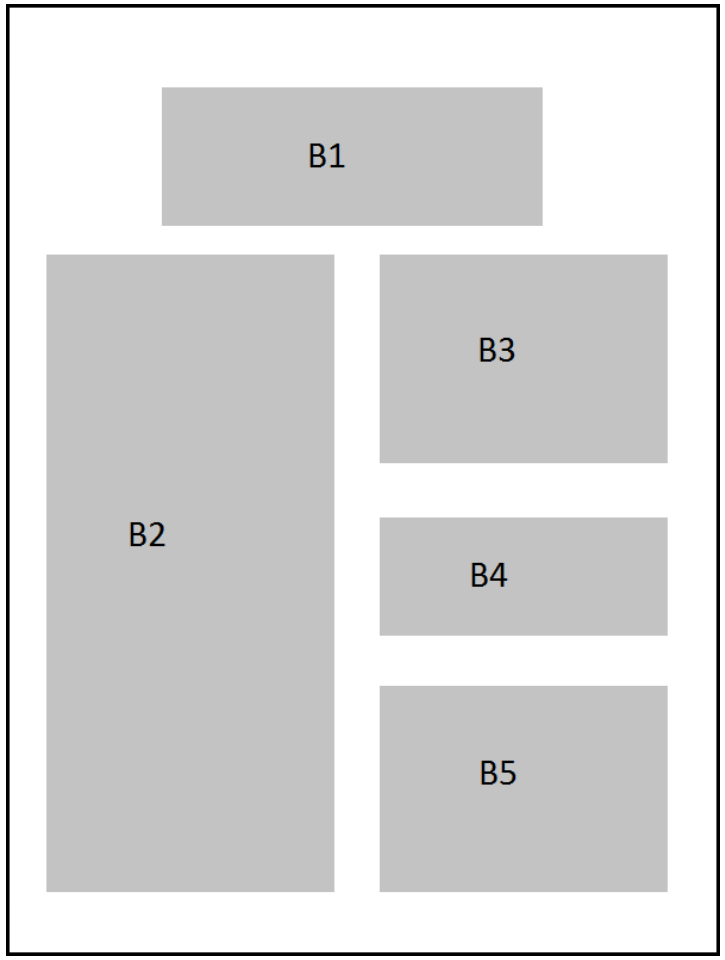
$(iv)\ W = X \setminus \bigcup_1^n B_j$ is called background and $\text{Prop.}(B) \neq \text{Prop.}(E_i)$

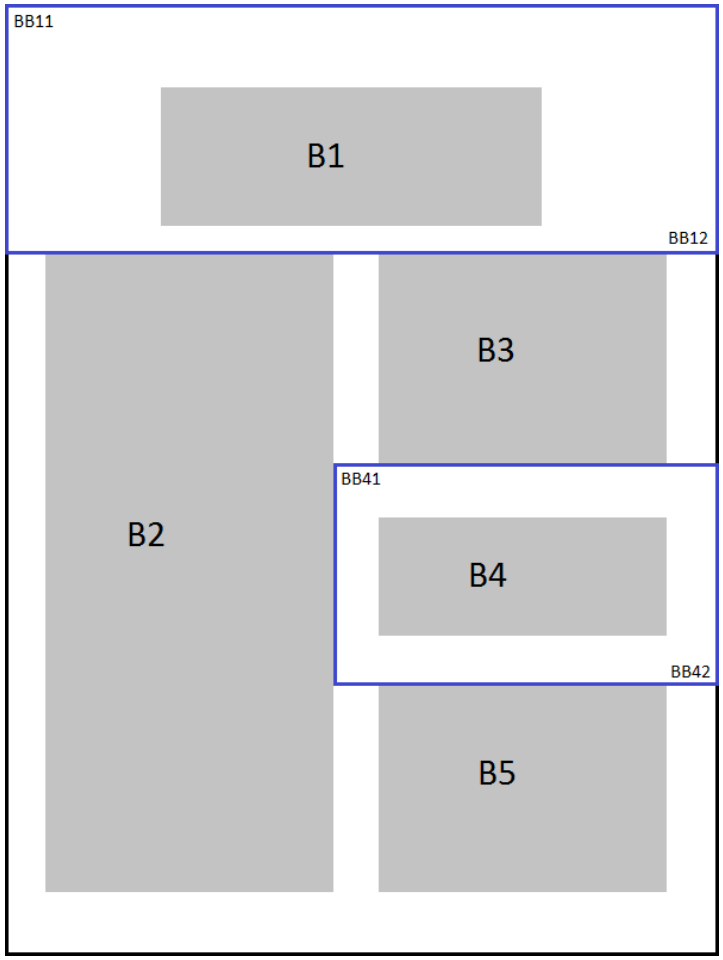
Performance evaluation



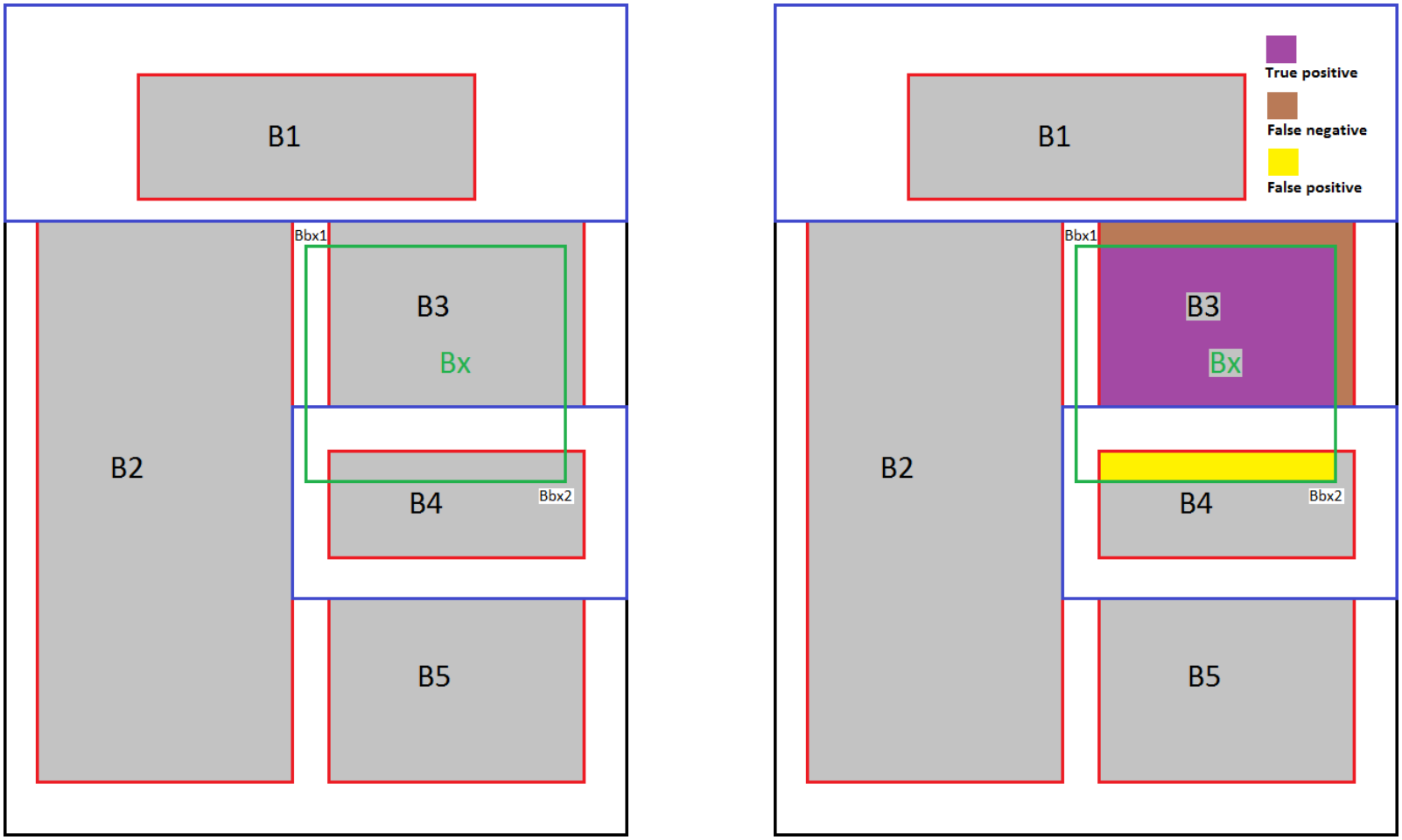


Performance evaluation

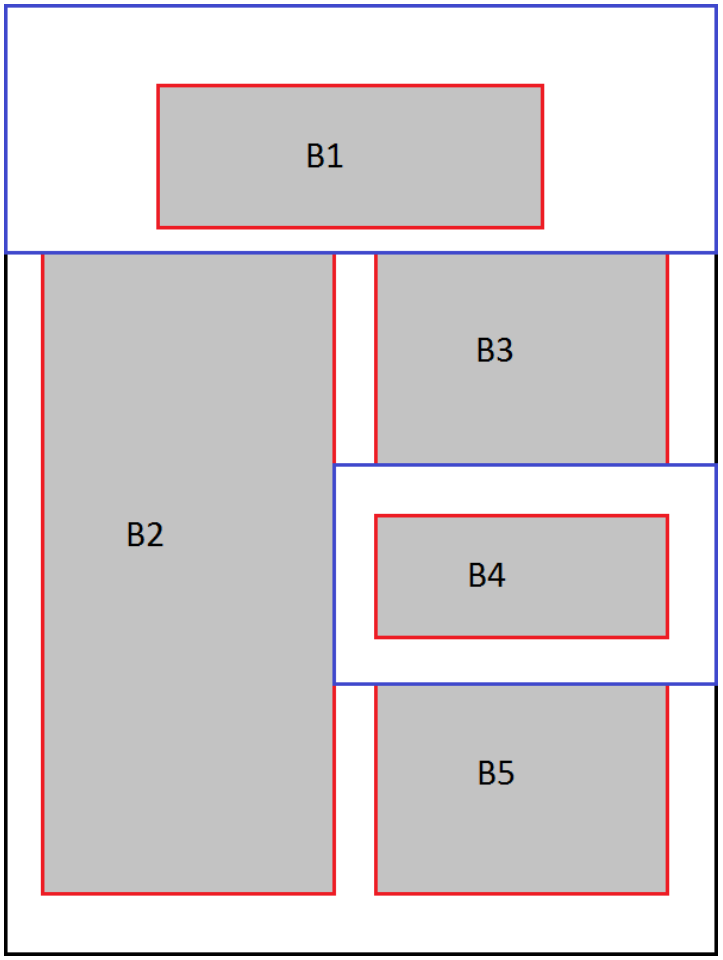


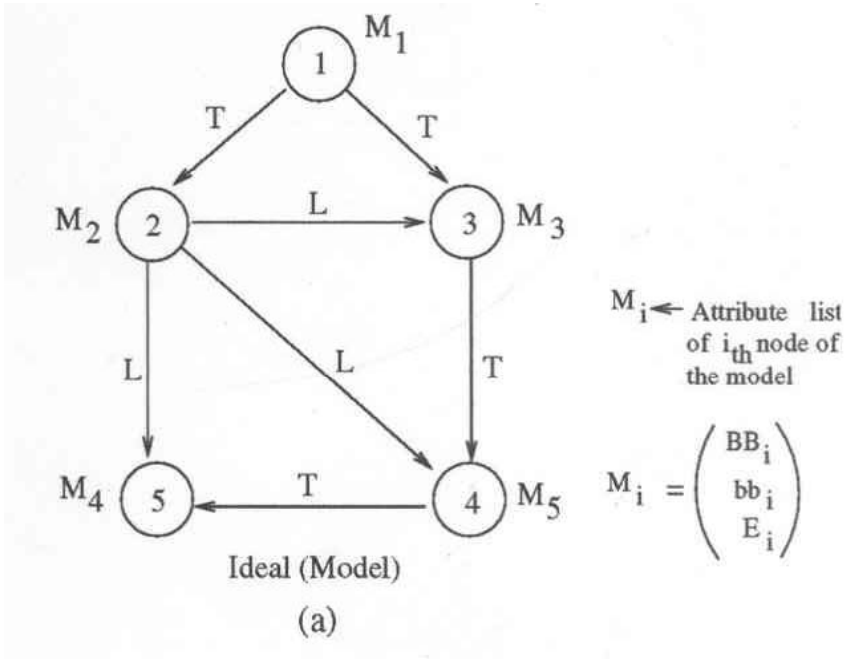


Performance evaluation



Performance evaluation





Performance evaluation

- Both model and object graphs are directed acyclic graph.
- Let us represent the model graph by

$$G_M = (V_M, L_M)$$

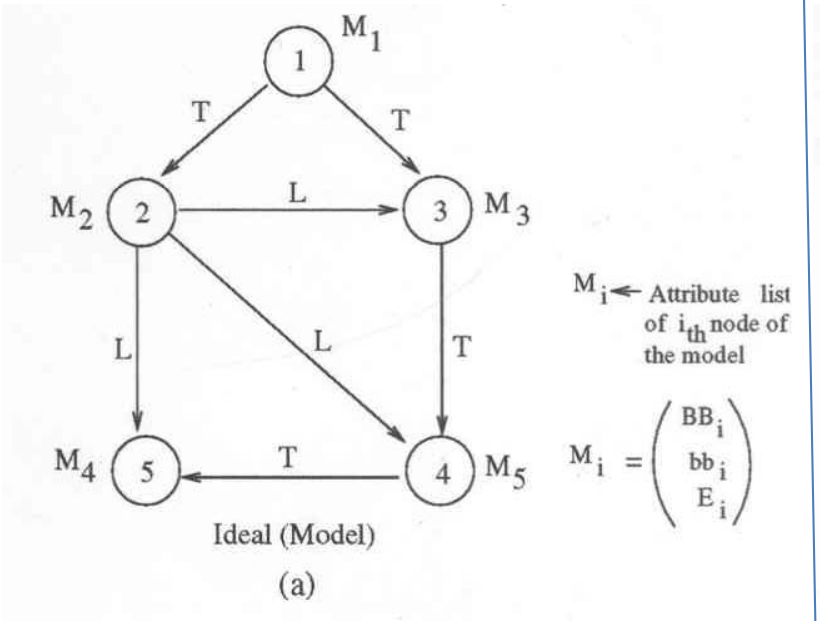
where $V_M = \{M_0, M_1, M_2, \ldots, M_n\}$ represents the set of nodes or vertices and L_M represents set of links.

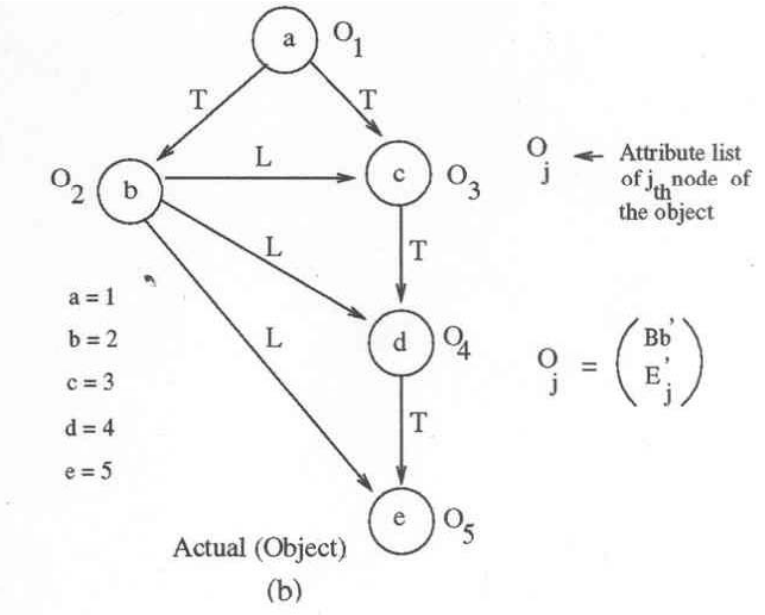
- Note that $M_j = (BB_j, bb_j, E_j)$ and $L_{jk} = (M_j, M_k)$.
- Similarly the object graph is represented by

$$G_o = (V_o, \mathscr{L}_o)$$

- And $O_j = (Bb_j, \mathscr{E}_j)$ and $\mathscr{L}_{jk} = (O_j, O_k)$.
- Finally, graph matching algorithm is employed.

Performance evaluation





Das, Saha and Chanda, An empirical measure of performance of document image segmentation algorithm, *IJDAR, Vol. 4(3), 2002.*

Performance evaluation

- Relation between BB_j and bb_j in model (groundtruth):

$$BB_j \supseteq bb_j \text{ and } BB_j \bigcap bb_k \text{ for } j \neq k$$

- For good segmentation of object node:

$$bb_j \subseteq Bb_i \subseteq BB_j \text{ if node } M_j \text{ matches node } O_i$$

- The error measure:

  (i) Correct classification *(True positive) = #($Bb_j \cap BB_i$).*

  (ii) False alarm *(False positive) = #($Bb_j \setminus BB_i$).*

  (iii) Mis-classification *(False negative) = #($bb_i \setminus Bb_j$).*

Conclusion

- Presented a document image segmentation method based on shape features
- Used mathematical morphological operators
- Necessary for OCR and data compression
- System is useful for development of digital library providing facilities for electronic storage, searching, navigation

References

- B. Chanda and D. Dutta Majumder, **Digital Image Processing and Analysis**, Prentice Hall of India, New Delhi, 2000.
- A. K. Das and B. Chanda, **A fast algorithm for skew detection of document images using morphology**, Intl. J. Of Document Analysis and Recognition}, Vol.4, pp.109-114, 2001.
- A. K. Das, S. K. Saha and B. Chanda, **An empirical measure of performance of document image segmentation algorithm**, Intl. J. on Document Analysis and Recognition, Vol.4, pp.183-190, 2002.
- S. Mandal, A. K. Das and B. Chanda, **A Simple and Effective Table Detection System from Document Images**, Int. J. on Document Analysis and Recognition, Vol.8, pp.172-182, 2006.

Thank you