

Text Recognition and Retrieval in Natural Scene Images

Thesis submitted in partial fulfillment
of the requirements for the degree of

MS
in
Computer Science

by

Udit Roy
201207725

`udit.roy@research.iiit.ac.in`



Center for Visual Information and Technology
International Institute of Information Technology
Hyderabad - 500 032, INDIA
December 2015

Copyright © Udit Roy, 2015

All Rights Reserved

International Institute of Information Technology
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled “Text Recognition and Retrieval in Natural Scene Images” by Udit Roy, has been carried out under my supervision and is not submitted elsewhere for a degree.

Date

Adviser: C V Jawahar

Date

Co-Adviser: Karteek Alahari

To my family

Acknowledgments

First of all, I would like to express my sincere gratitude to my advisor, C. V. Jawahar, for providing me with an opportunity to pursue research under his guidance with continuous support, patience and motivation. I extend my gratitude to my co-advisor, Karteek Alahari, for inculcating several research practices in me, especially critical analysis and presentation skills with a lot of patience. I thank both of them for teaching me time management, focused research and developing essential people skills which will be immensely beneficial in the long run.

I thank my mom and dad, younger brother, grandmothers, uncle and aunt for emotional support, love, trust and keeping my spirits up in difficult times. I express my deepest thanks to my collaborators, Naveen Sankaran, Pramod Sankar and Anand Mishra, who worked with me on various projects in the lab and helped me develop essential skills through discussions. I am thankful to my friends, Viresh, Devendra, Praveen, Vidyadhar, Natraj, Aniket and Pritish for helping me out by discussing ideas and reciprocating the same. I thank my friends, Tejaswinee, Pramod, Koustav, Saurabh and Rajvi for all the fun times we had as well as extending my knowledge to other fields too. I acknowledge the support of my other IIIT friends who helped me pursue various other life goals like fitness and sports. I am grateful to our CVIT lab staff Satya, Rajan, Phani and their team for helping me out in several instances ranging from paperwork to data annotation. I also extend my thanks to CVIT faculty, P. J. Narayanan, Anoop Namboodiri, Jayanthi Sivaswamy, Avinash Sharma and Vineet Gandhi for making the lab a good place to pursue research in computer vision and machine learning. Last but not the least, I convey my deepest thanks to Gaurav Harit for giving me a direction to IIIT in a crucial time; my journey here wouldn't have started without him.

Abstract

In the past few years, text in natural scene images has gained potential to be a key feature for content based retrieval. They can be extracted and used in search engines, providing relevant information about the images. Robust and efficient techniques from the document analysis and the vision community were borrowed to solve the challenge of digitizing text in such images in the wild. In this thesis, we address the common challenges towards scene text analysis by proposing novel solutions for the recognition and retrieval settings. We develop end to end pipelines which detect and recognize text, the two core challenges of scene text analysis.

For the detection task, we first study and categorize all major publications since 2000 based on their architecture. Broadening the scope of a detection method, we propose a fusion of two complementary styles of detection. The first method evaluates MSER clusters as text or non-text using an adaboost classifier. The method outperforms the other publicly available implementations on standard ICDAR 2011 and MRRC datasets. The second method generates text region proposals using a CNN based text/non-text classifier with high recall. We compare the method with other object region proposal algorithms on the ICDAR datasets and analyse our results. Leveraging on the high recall of the proposals, we fuse the two detection methods to obtain a flexible detection scheme.

For the recognition task, we propose a conditional random field based framework for recognizing word images. We model the character locations as nodes and the bigram interactions as the pairwise potentials. Observing that the interaction potentials computed using the large lexicon are less effective than the small lexicon setting, we propose an iterative method, which alternates between finding the most likely solution and refining the interaction potentials. We evaluate our method on public datasets and obtain nearly 15% improvement in recognition accuracy over baseline methods on the IIIT-5K word dataset with a large lexicon containing 0.5 million words. We also propose a text query based retrieval task for word images and evaluate retrieval performance in various settings.

Finally, we present two contrasting end to end recognition frameworks for scene text analysis on scene images. The first framework consists of text segmentation and a standard printed text OCR. The text segmented image is fed to Tesseract to get word regions and labels. This case sensitive and lexicon free approach performs at par with the other successful pipelines of the decade on the ICDAR 2003 dataset. The second framework combines the CNN based region proposal method with the CRF based recognizer with various lexicon sizes. Additionally, we also use the latter to retrieve scene images with text queries.

Contents

Chapter	Page
1 Introduction	1
1.1 Recognition of Printed Text	2
1.1.1 Brief Overview	2
1.1.2 Incompatibility with Scene Images	3
1.2 Understanding Scene Text	3
1.2.1 Characteristics	4
1.2.2 Challenges	6
1.3 Contributions	8
1.4 Thesis Outline	9
2 Scene Text Detection	10
2.1 Introduction	10
2.2 Prior Art	12
2.2.1 Early text detection (2000-2005)	12
2.2.2 Emergence of connected components based methods (2006-2010)	13
2.2.3 MSER based methods (2011 to present)	15
2.3 A flexible detection scheme	18
2.3.1 Precision or Recall: What is important in detection	19
2.3.2 Text Detection via Hierarchical Clustering	20
2.3.3 Text Detection via CNN Classifier	23
2.3.4 Fusion	24
2.4 Experimental Analysis	24
2.4.1 Datasets	25
2.4.2 Evaluation	25
2.4.3 Text Detection via Hierarchical Clustering	27
2.4.4 Text Detection via CNN Classifier	29
2.4.5 Fusion	31
2.5 Summary	33
3 Scene Text Recognition and Retrieval	34
3.1 Introduction	34
3.2 Related Work	35
3.3 Proposed Method	36
3.3.1 CRF framework	37
3.3.2 Generating Candidate Words	38

3.3.3	Diversity Preserving Inference	39
3.3.4	Lexicon Reduction	40
3.4	Recognition and Retrieval	41
3.5	Experimental Analysis	43
3.5.1	Datasets	43
3.5.2	Multiple Candidate Word Generation	43
3.5.3	Diversity Preserving Inference	44
3.5.4	Recognition	44
3.5.5	Retrieval	46
3.6	Summary	48
4	End to End Frameworks	49
4.1	Introduction	49
4.2	Related Work	49
4.3	Proposed Method	50
4.3.1	Text Segmentation + Tesseract	50
4.3.2	Region Proposals + CRF based Recognition	51
4.3.3	Scene Image Retrieval	51
4.4	Experimental Analysis	52
4.4.1	Datasets	52
4.4.2	Evaluation	52
4.4.3	Text Segmentation + Tesseract	52
4.4.4	Region Proposals + CRF based Recognition	52
4.4.5	Scene Image Retrieval	53
4.5	Summary	54
5	Conclusions	57
5.1	Future Work	58
	Bibliography	60

List of Figures

Figure	Page
1.1 Typical printed text image (a) vs. scene image (b)	4
1.2 Foreground variations in scene text	5
1.3 Typical background variations in scene images	6
1.4 Detection styles recognized by the community	7
2.1 Typical detection pipeline followed in 2000-2005. Method by Ye et al. [131]	14
2.2 Stroke width transform pipeline by Epshtein et al. [24]	15
2.3 Overview of a robust text detection method by Yin et al. [132]	18
2.4 Text detection pipeline	20
2.5 Overview of single linkage clustering on MSERs	21
2.6 CNN based text region proposal on a scene image	23
2.7 MSRA dataset evaluation protocol	26
2.8 Precision recall curves for the classifier based methods on the ICDAR 2003 dataset . .	30
2.9 Qualitative results for the SLC + adaboost method on images from ICDAR 2003 dataset	30
2.10 Precision recall curve for various proposal methods	32
2.11 Text detection performance of the fusion	33
3.1 Examples where the diverse solutions contain the correct result	35
3.2 Overview of the proposed framework	37
3.3 Visualization of candidate words generation on a sample image	38
3.4 Qualitative results where retrieval results are correct	47
3.5 Failure cases for retrieval experiment	47
4.1 End to end pipeline for text segmentation + Tesseract	51
4.2 Lexicon based end to end framework performance of various methods	54
4.3 Successful qualitative results for the SLC + Tesseract method	55
4.4 Qualitative scene image retrieval results	56

List of Tables

Table	Page
2.1 Summary of text detection methods from 2000-2005	14
2.2 Summary of text detection methods from 2006-2010	16
2.3 Summary of text detection methods from 2011 to present	19
2.4 Text Detection Datasets	25
2.5 List of text detection implementations used for evaluation	28
2.6 Text detection results of various methods on the popular datasets	31
2.7 Text region proposal results of various methods on ICDAR datasets	32
3.1 Group edit distance example	41
3.2 Effect of the lexicon reduction technique on the inferred label	44
3.3 Word recognition accuracy comparison between various CRF and non-CRF methods .	45
3.4 Word recognition accuracy comparison while varying number of solutions	46
3.5 Top-1 precision for retrieval experiment on various datasets	48
4.1 Lexicon free end to end framework qualitative performance	53

Chapter 1

Introduction

In the last decade, we saw a surge in multimedia content generated across the world. With the arrival of digital equipments like cameras, camcorders, etc., it became feasible for a large section of the world's population to generate such content. These devices became even more popular especially after their improvement in performance, gradual decrease in prices as well as their integration into cell phones. In fact, after the increase in mobility of such devices via cellphones, coupled with the advances in cheap storage, it was possible to capture media content on the fly in the form of images and videos.

The multimedia content generated at large scale by the population provided us the opportunity to tag, categorize and make them browsable [37,71,95,115]. This scenario gained more significance when the content was uploadable to the internet where millions of people can access them. Human driven tasks like annotating the content to identifying people in a security camera feed became challenging as the amount of content grew. To compensate that, automated systems were developed to mimic human perception of the content. Several use cases came into the picture such as, (i) annotation systems which tagged the image and videos based on various properties and its content, (ii) retrieval systems which utilized such annotations to develop scalable solutions to index the content and, (ii) real time systems to extract certain kind of information e.g., understanding the surroundings in case of navigation, analysing sports videos, etc.

Several kinds of information can be extracted from multimedia content with varying levels of computation [110]. At the the lowest level we have trivial information which require no computation like meta-data provided with the image (e.g., date of picture taken, camera model etc.) [65]. With a little computation, we can perform some simple processing and generate tags from text associated with the image (e.g., image name and description) or also around the text in case of web pages via keywords finding or text summarization techniques [35, 122]. At the highest levels of computation, we try to interpret the image cognitively were we 'look' into the image and identify objects, persons, places, etc. in them [61, 108]. We see that as the higher levels of computation gives us more relevant information as compared to its lower counterparts, making the information extraction process more useful. For example, from a holiday picture, it would be most informative to tag the image with number of people, their

facial expressions and objects around them than with text associated which describe the image as family on vacation or the date and location where the picture was taken.

In this thesis, we work on one such taggable content in images called scene text, which refers to text present in the images in the form of advertisements on billboards, shop names in market squares, informative signs on boards, building names etc. We work on solutions to the challenge of digitizing the text in images, addressing one kind of information from the multimedia content.

1.1 Recognition of Printed Text

Before the surge in multimedia content, we advanced in printing technologies as a medium of mass communication. Information and thoughts were shared using books, newspapers, magazines, etc. Such content even though designed to reach a larger community, was insufficient in reaching throughout the country or world due to (i) utilization of resources like paper, ink, wood, transportation, etc. which were infeasible for such a large scale production (ii) searchability within such content as it grew in size, and (iii) language barriers. However, as the processing power of computers increased in the 90s, the document analysis community was formed to address the above challenges by digitizing the content thus, making it easily shareable over internet and enabling language translation on them.

1.1.1 Brief Overview

The document analysis community dealt with problems specific to digitizing printed text content [20, 56, 80]. With this common objective, several sub-problems were explored by the community. Documents in the form of books, magazines, old historic documents, etc. were the source of printed text. The documents were scanned page by page into high resolution images. These scanned images were generally pre-processed to remove noise and provide a cleaner image for further processing. The text was segmented into individual lines and words using various layout identification techniques. These segmentations were then used to recognize the content and building retrieval systems to search within them.

For pre-processing, image processing techniques considering the image as a signal were utilized. The documents required to be binarized to separate text from its usually simple background. Methods like Otsu's global thresholding method [88] to local thresholding methods of Sauvola [96] were developed. Noise like salt and pepper, ink bleeding, etc. due to poor scanning or artefacts were also needed to be removed [3, 15]. Documents were also enhanced during the binarization step [27, 119]. After binarization, the layout of the text was identified from the foreground and lines and words were segmented out [50, 68, 87]. Text in a page was separated into paragraphs, lines and words by searching for text baselines and whitespaces. Additional content in the page like page numbers, pictorial content in the form of graphs, diagrams, pictures, etc. were located and kept out of the processing loop.

The segmented words were now recognized by either recognizing character one by one or using a holistic approach [93]. Methods targeting different languages like arabic [4, 62] to various Indic scripts [34, 89] were also addressed. Optical Character Recognition (OCR) systems were built which clubbed all parts of document analysis and presented an end to end solution. Some notable systems are Tesseract [109] and ABBYY which can recognize text from scanned document images in real time with support for multiple languages. Documents were also indexed for retrieval [20]. Concepts like query expansion for relevant retrieval were also explored [127]. Robust methods with tolerance to character recognition errors were also developed [69].

Offline and online handwritten documents possessed a different set of challenges which were addressed too [10, 91, 93, 111]. Online handwriting recognition methods have evolved extensively and have started appearing in various software products. Microsoft Office's handwriting to text conversion tool and Analog Keyboard Project ¹ and Google's Handwriting Input application are a few examples.

1.1.2 Incompatibility with Scene Images

The methods devised by the document analysis community mostly exploited the patterns found in printed text. In printed text we come across characteristic features like dominance of text content in a page, standard layouts, adherence to a single font style, simple black on white text etc. which can only be found in printed text. However, in scene images we see that text is sparse and can be found in any style with varying foreground and background color complexities. They follow layouts which can be cognitively perceivable by humans thus generating a large scope of possibilities as compared to a limited layout set present in printed text content. In Figure 1.1(a) we see a typical scanned book page with a fixed layout consisting of page number, book title and paragraphs with even line spacing and font type. The consistency can be exploited and the same method can be applied to the other pages of the book to recognize the text. On the other hand, the scene image consists of text with varying sizes with a layout that is only applicable to this particular image. Owing to this variation/non-uniformity present in scene text, a direct application of OCRs were expected to perform poorly on them, thus requiring alternative strategies to address the challenges.

1.2 Understanding Scene Text

Multimedia content generally consists of images and videos taken in wild with no prior restriction on their content. We generally refer to them as scene images signifying the the content possesses various kinds of scenic information in them. The text present in them is eventually referred to as scene text.

¹<http://research.microsoft.com/en-us/um/redmond/projects/analogkeyboard/>



Figure 1.1 Typical printed text image (a) vs. scene image (b). Printed text are generally scanned or camera acquired and have a simple foreground and background style following standard layouts of newspapers, books, etc. Headings and para text are clearly distinguishable with pictorial content like images are wrapped by text with sufficient space. On the other hand, text in scene images are pictures taken using cameras and may contain a lot of non-text content. The text itself may have complex foreground and/or background with a varied layout pattern. Text in a single image may not observe similarity in terms of font styles and sizes as well as can be distorted.

1.2.1 Characteristics

As mentioned above, unlike its printed counterpart, scene text possesses several varying features which make it more challenging to deal with. Its properties can be categorized in the following groups which help us understand its complexity,

- **Foreground** - This property corresponds to the features used to create the characters and words in the text. It includes the following major font properties,
 - **Style** - Can range from simpler fonts to handwriting style fonts which make it difficult to separate characters. For example, Figure 1.2(a) shows two images with contrasting font styles. On one hand, the characters are simple and clearly separated from each other, while the text “Diet Coke” has some perspective distortion with characters touching horizontally and vertically.
 - **Size** - Text in images can acquire any percentage of image area depending on zoom level of the image taken.
 - **Spacing** - Inter-character and inter-word spacings coupled with the size property can make it increasingly challenging to segment each element. Figure 1.2(c) shows an image with text

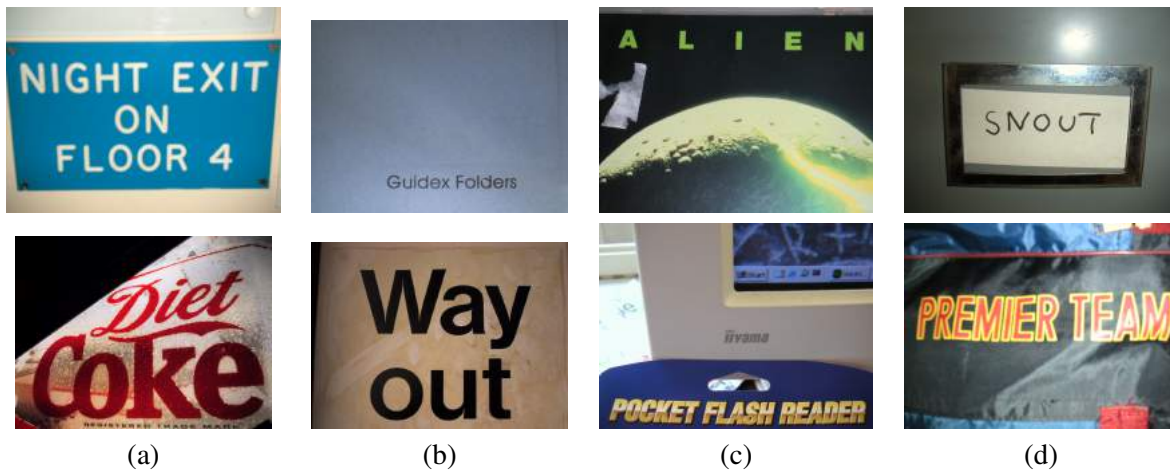


Figure 1.2 Foreground variations in scene text. (a) font styles from simple sans-serif type to handwritten (b) increasing text size w.r.t. image size (c) Text spacing from widely spaced to touching characters (d) Single color text to multiple color text with noise

“POCKET FLASH READER” which has no inter-character spacing while the other text in the same image utilize different font style overall. This complicates the search for text on a single image too.

- Colors - Text can possess varying color combinations ranging from single color foreground to several colors used in a single stroke.

We also observe that the foreground may not adhere to all font properties, i.e. for each character or word, there can be any combination of such properties thus drastically increasing its complexity. Figure 1.2 shows typical scene images with the above mentioned variations in foreground.

- **Background** - The text generally possesses a background which can be even more challenging to guess. It can broadly be divided into two categories,
 - Textured - Any pattern which can be considered as a single element being repeated several times falls into this category. It consists of solid color background to patterned background with multiple colors. Figure 1.3(a) shows us a typical example where the background is solid coloured board. The text can be extracted in an easier way as locally the image becomes similar to a printed text document.
 - Non-Textured - Cases where the background has no repeating element like background arising from text on transparent glass, etc. fall into this category. These backgrounds may also intervene with the foreground thus adding artefacts which distort or merge or over-split text thus making it far more challenging for us to digitize. Figure 1.3(b) has transparent text over a background image, which makes it difficult to separate the text, even locally.

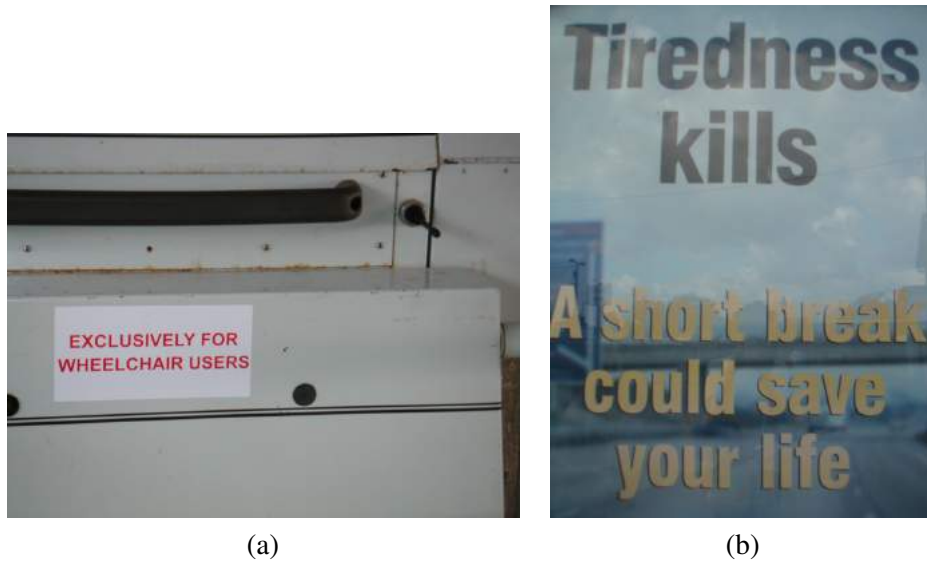


Figure 1.3 Typical background variations in scene images, (a) text with solid color background and, (b) text over a scene image providing a non-textured background with no repeating pattern

1.2.2 Challenges

Broadly speaking, scene text analysis includes locating and identifying the text content in scene images [43, 56, 136]. Since 2003, the International Conference on Document Analysis and Recognition (ICDAR) organizes a Robust Reading Competition [45, 63, 64, 97] where the state-of-the-art methods are presented. Accordingly, following are the major challenges categorized by the scene text community,

1. **Text Detection** - The objective of such methods is to locate the exact region in the scene image with the text. In other words, its task is to find a minimum sized region of interest with all of the text in the image inside it. The minimum sized region is generally considered in two forms, (i) a set of rectangular bounding boxes locating the text lines or words in a box [64] or (ii) segmentation with text pixels as foreground in a binary image [16]. Hence a typical system takes in a scene image as input and outputs a set of non-overlapping bounding boxes or a single binary image as shown in Figure 1.4. Such systems are evaluated on the basis of precision and recall computed using the intersection upon union statistics with respect to the ground truth. Both the detection methods are contrasting as,

- In detection via bounding boxes, word/line level bounding boxes are desired while in segmentation techniques character grouping information is absent.
- Segmentation methods also separate the foreground from background too which might be helpful in identifying the text. In case of detection via bounding boxes, the text still has background in the bounding boxes.



Figure 1.4 The two detection styles recognized by the community. Images in (a) show the detection results by Epshtein et al. [24] where text is located in yellow rectangles, and (b) shows images and their text segmentations by the method in [30].

2. **Text Recognition** - Once the text region is located, the region can be cropped and processed further to recognize the text. We implicitly separate the foreground from the background and then recognize the text. Recognition can be done character wise or holistic with or without the help of a language model. They are evaluated based on the categorical correctness of the predicted characters with respect to the ground truth word. Since the text present in scene images may have (i) out of dictionary words and, (ii) high amounts of variation leading to character misclassification, unconstrained scene text recognition is also approximated by a lexicon based approach.

The lexicon based scene text recognition consists of recognizing words in a cropped image with the ground truth present in a provided list of words referred as a lexicon. Hence, the objective of recognition now narrows down to selecting a word from the lexicon which is highly likely to appear in the image. The lexicon based approach is useful in the setting where we know the image is definitely supposed to contain text from the lexicon. The accuracy is usually inversely proportional to the lexicon size and reaches the performance of unconstrained recognition as it increases in size.

3. **Text Region Proposal** - A variation of the text detection challenge, here our objective is find a set of bounding boxes with very high text recall. Typically these methods are used in conjunction with a strong recognition system which removes the false positives from the proposed regions, thus providing a feedback. These methods are ideally supposed to be faster than a detection system, generate bounding boxes with very high text recall and a feasible number of regions.

4. **End to end frameworks** - Just like OCRs in printed text, end to end frameworks propose a localization and recognition solution together. Given an image as an input, its objective is to locate the text and recognize it. Two major categories of such frameworks exist, (i) forward only approach with a detection and a recognition module or, (ii) feedback based approach with a text region proposal method and a recognition module which simultaneously recognizes text and reduces proposals to correct detections.
5. **Retrieval systems** - In this scenario, we intend to retrieve images with the corresponding text query from a database. The database can consist of scene images in case of a retrieval system equipped with a end to end framework. Another interesting setting is when the detection step is assumed to be done resulting in a database of cropped word images. As in recognition, the retrieval systems can also be built using lexicons thus restricting query space to the lexicons only. These methods are evaluated by the mean average precision metric using a specific set of query text.

1.3 Contributions

In this thesis, we address the challenges of text detection and recognition. We propose solutions to detection via segmentation and a recognition with the help of a lexicon. The methods are combined to form full end to end recognition pipelines. Retrieval systems for scene images as well as word images are also explored using the methods. Our contribution in this thesis is multi-fold and stated as follows,

1. We study how the text detection challenge is addressed by the scene text community in the past two decades. We categorize their trends in terms of techniques using ranging from image processing methods to computer vision and machine learning based approaches. We highlight the common pipeline architectures and discuss their merits and demerits. Most important works are discussed in a chronological order.
2. We propose a fusion of two complementary styles of detection,
 - A text detection via segmentation method which is capable of detecting multi-script text in multi-orientation setting. We compare the method with other publicly available methods and show that it performs at par on the ICDAR 2011 [97](containing horizontal English text) and MRRC [53](multilingual with text in arbitrary orientation) datasets.
 - A text region proposal method using a patch based CNN text/non-text classifier is discussed and evaluated against seven other object region proposal methods. They are compared on the basis of text recall, average proposals and time taken.
3. We propose recognition method is for cropped word images with a specific lexicon based on conditional random fields. We alternate between inferring the best diverse solutions and reducing

the lexicon to make the pairwise interactions meaningful. We evaluate our approach against other state of the art methods and find it to outperform them by 15% on the IIIT-5K dataset with a large lexicon comprising of half a million words. This is the primary contribution of the thesis.

4. We propose a solution to query by text retrieval of cropped word images with a specific lexicon. We reduce the lexicon and use it to index the cropped word images for retrieval. Experiments are performed with diverse and non-diverse solutions with various settings on standard datasets.
5. The first end to end pipeline comprising of a text detection via segmentation and the Tesseract OCR is evaluated. We find its performance to be at par with other important pipelines with case sensitive and lexicon free recognition and detection.
6. The second end to end pipeline comprising of the CNN based region proposal method and the CRF based recognition module is built. Their performance using two different lexicon sizes are evaluated and compared with other state of the art methods. A text to scene image retrieval application is also developed using this pipeline. We find it to be outperformed by them and justify it in our discussion.

1.4 Thesis Outline

In this thesis, we first discuss the text detection methods in chapter 2. The various works addressing the text detection challenge by the community are evaluated and categorized in the comprehensive survey section 2.2. We then present a text segmentation method, a region proposal method and propose a fusion of these methods in the proposed methods section 2.3. The experiment section 2.4 evaluates the methods on various datasets as well as shows qualitative results with insights.

In chapter 3, a lexicon based scene text recognition method is presented. Section 3.2 discusses the related works to our recognition scheme. The proposed pipeline common to recognition and retrieval is described in section 3.3. The recognition and retrieval specific setting for the pipeline is described in section 3.4. Experimental analysis is presented in section 3.5 with quantitative comparisons and qualitative results.

Chapter 4 presents two contrasting frameworks based on the detection methods described in chapter 2. With the help of an OCR and the recognition engine described in chapter 3, we formulate two end to end recognition pipelines which are described in section 4.3. Experimental results are shown in section 4.4 with a few qualitative results and relevant analysis.

The last chapter concludes and discusses relevant future work in scene text understanding.

Chapter 2

Scene Text Detection

2.1 Introduction

Text detection is an important part of scene understanding. Its objective is to find a set of sub-regions in the image with text and minimal non-text content. In general, text detection can be done in three ways,

1. Text detection by locating text in bounding boxes (best fit or horizontal) [24, 44, 82, 83, 85, 100]
2. Text extraction by binarizing the scene image such that all text pixels are foreground and the rest are background [30, 31]
3. Text region proposals methods giving multiple possible text bounding boxes [39, 40]

While categories 1 and 2 relate to explicit location of text, methods in the third category depend on the a recognition module to correctly locate the text. Region proposals are advantageous in developing end to end frameworks and retrieval systems due to high recall in detection, allowing fewer errors to propagate further.

In the last two decades, several approaches to text detection have been pursued. Earlier pipelines relied on image processing techniques to get text pixels which were then merged together to form regions. Text detection was done for captions in video frames which were subsequently applied to scene images. In an effort to standardize and consolidate the progress, the ICDAR Robust Reading Competitions 2003-2013 [45, 63, 64, 97] were organized eventually, providing standard datasets and evaluation protocols. The f-measure of the best text detection systems increased from 50% to 75%, showing substantial progress. However, the effort by the community still remained on detecting English text in horizontal orientation which is only be treated as a sub-task for multi-script multi-orientation detection.

The text detection methods of this decade mostly rely on a common framework. The pipeline starts with identifying indivisible components which can correspond to characters and then group or merge them to words. Typically, components are extracted as maximally stable extremal regions [70] or stroke width components [24] and classified as character or non-character. These components work well in

case of English where in most of the cases, the alphabets can be grouped pixel-wise in the image except for the case of handwriting which have the whole word as a component. The character candidates are now merged to form words, relying on word formation rules of English script. Such frameworks are slightly disadvantageous as (i) the character classification is language specific and comes early in the pipeline, hence errors can be propagated or incremented further (ii) no n-gram level information is utilized which can serve as a script independent important cue in increasing confidence of the true positives, along with better rejection of false positives and, (iii) reliance on heuristics for rejection of falsely classified word regions.

An alternative framework with the capability of detecting multi-lingual and multi-orientation text soon came into picture which addressed the earlier mentioned demerits. The new framework had grouping of the indivisible components first, followed by a supervised classification step. The grouping stage is done earlier with the intention to merge similar components or characters without the knowledge of the script or orientation. The clusters formed can then be classified with much more robustness as compared to single character classification while still maintaining script and orientation independence. Typically these frameworks use the hierarchical clustering technique on these components and then classify clusters as text or non-text using hand-crafted features which are independent of any language. The features rely on coherence and similarity of the components which serve to be a very important script independent cue. At last, the text clusters are merged using a simple operation like union and further processed to get bounding boxes or text segmentations.

More recently, text was treated as object and following the style of top performing Pascal VOC object detectors [36], text region proposals were generated as an alternative to sliding window based search of text region. The proposals can later be classified and retained if they contain text. Following this style of detection, various object region proposal methods can be used to find the salient regions with text. However since text may not be considered as a category or instance level object, due to lack of visual structure in a word as well as distinguishable parts, an effective strategy is required to detect their presence. One way to perceive it was as a collection of characters as objects, obeying the Gestalt laws of grouping [46]. A plausible solution can be to slide a patch based text/non-text classifier to generate a character confidence map and then group the high scoring regions to bounding boxes. Owing to the recent success of Convolutional Neural Network (CNN) based classifiers in the scene text domain, we utilize a CNN classifier which evaluates all patches in the image to give us a score map with pixel level confidences for text presence. The score map is further processed to obtain bounding boxes with text and the process is repeated at several scales. This generates several bounding boxes with high recall which can be further processed by the recognition module.

In this chapter, we provide a survey of the evolution of the various detection methods published during the last 15 years in section 2.2. We categorize the major works done in this field till now while analysing the common techniques used during those specific years. We identify the major pipelines and cluster the surveyed publications, thus giving a bird's eye view to the reader. In section 2.3, we propose a flexible detection scheme by fusing a high precision text detection method with a high recall

text proposal method. The detection uses the hierarchical clustering on MSERs and an adaboost based text/non-text classifier. This pipeline’s effectiveness is evaluated on multi-lingual and multi-orientation text datasets and compared with other publicly available methods. The text region proposal method uses a CNN classifier to classify pixels as text or non-text at various scales. Several proposals as bounding boxes are generated and compared with other generic object proposal methods on the ICDAR datasets. We show the quantitative and qualitative results in section 2.4 followed by the summary in section 2.5.

2.2 Prior Art

In the last two decades, several methods have been proposed to detect text on videos and images in the form of captions, logos and signs. Methods were motivated by the advances in image processing and vision community in their respective time periods. Development of scalable and robust machine learning algorithms also contributed in improving performance and making such methods applicable to a larger portion of such images. Methods can be categorized mainly in three periods, each relying on key developments in the community. One such key development was MSERs [70] in the year 2004 which marked a significant change in approach to the detection challenge. In the years 2000-2005, methods lacked the connected component approach which was mainly introduced by the MSERs. Although MSERs were conceptualized in 2004, their extensive usefulness in scene text detection was identified only after 2010. In the years 2006-2010, the image processing based methods still worked on caption text from videos as in the earlier years. One major contribution came in the form of stroke width transform [24] which proposed a novel idea of forming connected components using stroke width from the image. Several datasets were publicly released to standardize performance comparisons within the community. Thereafter, from 2011 onwards, several works incorporated MSERs into their pipeline, greatly improving robustness for character detection. Deep learning based methods were also brought into the picture in the last few years, significantly improving classification performance.

2.2.1 Early text detection (2000-2005)

In this section, we review the important works in text detection prior to the development of methods using MSERs. Several surveys have also been done for detection which can be found in [21,43,56,134]. The key point in the methods from this era are reliance on *unsupervised image processing techniques* which were used with certain assumptions on the quality of text present. The methods assumed text to be contained in high contrast regions, generating edges of high intensity and single colored ensuring the respond uniformly to the image processing techniques employed. These methods, generally classified as *texture based*, had a pipeline which started with a *segmentation* step, responsible for generating a image with text areas having higher intensities. It was followed by a *detection* step which grouped the high intensity regions. The final step was a *verification* step which classified the candidate regions as text or non-text.

The *segmentation* step relied on image processing concepts like wavelets [28, 54, 59, 131] which assumed text to be belonging to the high frequency bands to edge features [11–13, 66, 123, 125]. Edge intensity and a high local density served as an important cue in locating text regions which was used to perform a coarse to fine grained detection in [11, 131]. Gradients were enhanced using simple filters to obtain high intensity text regions [123, 125]. The *detection* step applied simple morphological operations like dilation [12, 123, 125, 135] or opening [25, 131] to slightly complex methods like CAMSHIFT [48], etc. The gradient image was binarized and dilated using a horizontal filter in [123, 125]. The edge image was dilated horizontally and vertically and merged to get a single detection in [12]. The *verification* step was mostly heuristic based [66, 123, 125, 135] rejecting detected regions based on size, aspect ratio, edge densities, etc. Text/non-text classification with a SVM or a simple neural network classifier were also utilized in some of the works [14, 54]. In the experiment stage, most of these methods relied on demonstrating results on *caption text* in video frames which were found in abundance. It was in the year 2003, the ICDAR Robust Reading Competition [64] started, dedicating a separate challenge for scene text images. The methods which participated in the competition were mostly unpublished and lacked description in the competition report. However, one of the methods from the competition was published [123, 125], demonstrating a text detection and tracking system using a gradient based approach with dilation to find text candidates. Multiple frame integration was used to track and verify the detected regions. The ICDAR Robust Reading Competition 2005 [63] presented methods with superior performance and started to rely on connected component based approaches. The text detection challenge was won by Hinnerk Becker System which used an adaptive binarization technique to generate character regions. They were combined to form text lines using geometrical constraints and then classified as text or non-text based on histogram and edge features. Summarizing this time period, Table 2.1 highlights the key components used in the important works in a tabular form. The intermediate results from a popular wavelet based method by Ye et al. [131] in Figure 2.1.

2.2.2 Emergence of connected components based methods (2006-2010)

These years mostly focussed on text in videos in the form of captions and relied on image processing techniques. In the later years, few works started relying on connected components (CCs) which served as a precursor to the recent MSER based methods. Several incremental works were done by Shivakumara et al. [90, 102, 104, 105, 107] on caption text in video. The segmentation methods relied on laplacian output [90], wavelets [104, 107] or multiple filters [102]. The detection methods relied on simple pixel grouping techniques ranging from morphological techniques to block based approaches [102, 103] followed by heuristic based false positive elimination. Some preliminary connected component based methods were exploited by Kim et al. [47, 49] where image was converted into a binary image and components were grouped to text lines. The concept of stroke width as a filter design for text extraction was proposed in [60] which was later applied to caption text in [19, 42]. A highly successful work emerged in the form of stroke width transform by Epshtein et al. [24](Figure 2.2) with robustness and applications towards scene text. The stroke width transform was a two pass algorithm which outputs a

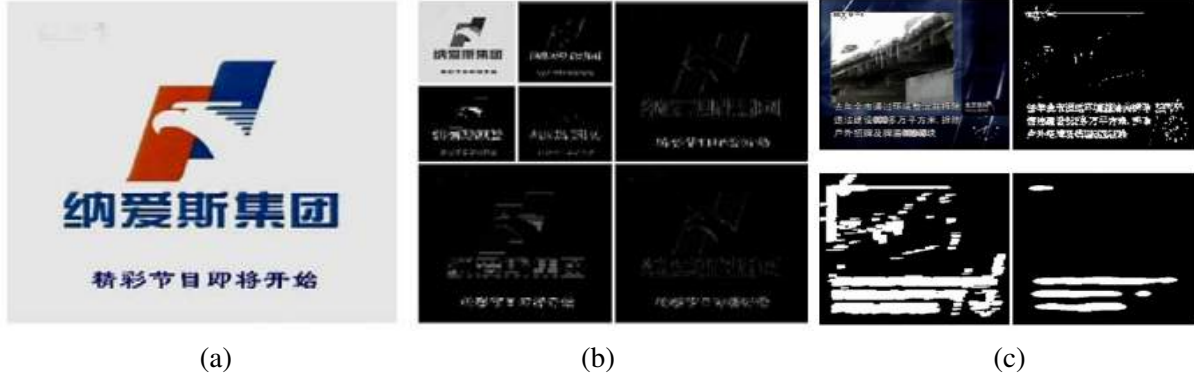


Figure 2.1 Method by Ye et al. [131]. (b) shows the two level wavelet decomposition of a sample image shown in (a). Each scale shows the LH,HL and HH decomposition with text in high intensity regions, suggesting text to lie in higher frequency regions. Row-wise from top left, (c) shows the original image, text pixels after wavelet decomposition and k-means clustering, horizontal candidate regions by morphological close operation, and density based region growing output to get final text regions.

Method	Segmentation	Detection	Verification
Li et al. [54]	wavelets	morphological	NN classifier
Zhong et al. [135]	DCT features	morphological	heuristics
Lienhart et al. [57, 58]	color based region growing + contrast based segmentation	geometry + texture + motion analysis	
Cai et al. [11]	selective local thresholding	morphological	heuristics
Wolf et al. [123, 125]	gradients	morphological	heuristics
Kim et al. [48]	text/non-text SVM classifier on patches	adaptive mean shift	heuristics
Chen et al. [13]	LoG edge detector	edge patch classification	
Gllavata et al. [28] & Liu et al. [59]	wavelets	unsupervised pixel block classification via k-means	heuristics
Chen et al. [12]	edge detectors	morphological	text/non-text MLP + SVM classifier
Ye et al. [131]	wavelets	morphological	text/non-text SVM classifier
Lyu et al. [66]	edge detectors	morphological	heuristics

Table 2.1 Text detection methods from 2000-2005. These methods generally show results on caption text like images from videos, assuming text to consist of simple foreground in a complex background setting. Typical pipelines consisted of a *segmentation* step to separate text like pixels, a *detection* step which groups such pixels and a *verification* step to reject false positives. We observe that most of the components used in such methods rely on image processing techniques and require heuristics. Unsupervised learning based approaches are also used extensively.

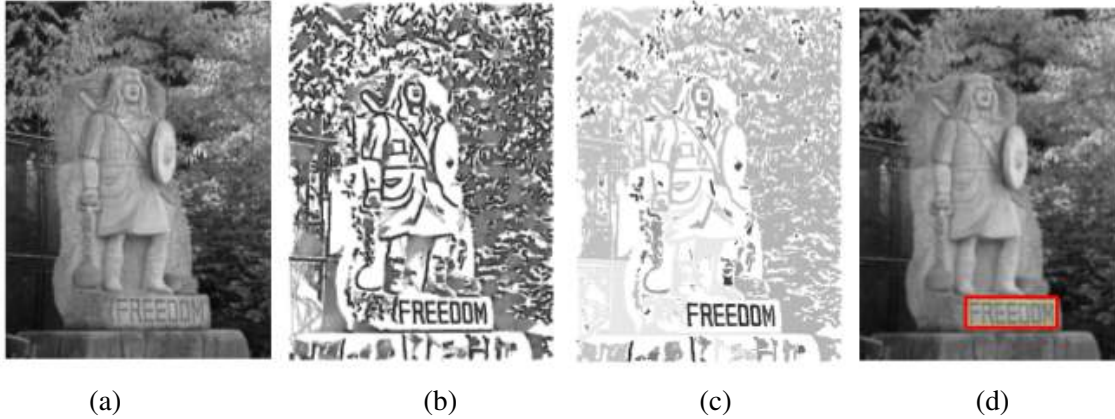


Figure 2.2 Stroke width transform method by Epshtein et al. [24]. Stroke width transform output (b) of image in (a). Each pixel stores the width of the stroke it is likely to be a part of. (c) shows the stroke width variance among the CCs relying on the fact that text components tend to have similar stroke width. (d) shows the final result on the image.

stroke width map storing the possible stroke width at each pixel. Since text is believed to be consistent in stroke width, CCs were extracted as character candidates. Candidates were then grouped on the basis of similarity with each other, leading to word and line level text detection. Overall, during this period, the scene text community did not receive much attention due to (i) limitation of image processing techniques, (ii) lack of standard datasets and evaluation protocols prohibiting comparisons and, (iii) absence of the Robust Reading Competition in ICDAR 2007 and 2009. A summary of important works during this period is presented in Table 2.2.

2.2.3 MSER based methods (2011 to present)

In the past few years, several scene text detection methods were proposed by Neumann et al. [82,83] which were highly successful due to MSERs [70]. The concept of MSERs i.e. connected components on grayscale image was exploited to generate candidate character regions robustly. The MSERs became a higher level operation on images in contrast to the low level image processing based methods used earlier. The standard framework consisting of a *segmentation* step was replaced by *character generation* step, thus moving from a texture based approach which classified as a pixel/patch as text/non-text to a CC based approach which finds possible characters on the image. The *detection* step which grouped classified pixels to text blocks were replaced by the *grouping* step, where the MSERs were grouped based on various similarity measuring properties into text lines. The *verification* step, however became complex utilizing better features either extracted from CCs or the image along with superior classifiers trained on huge amounts of data. The *verification* was also performed at CC level, classifying them using a character classifier to a word level where text blocks on image are classified as text or non-text. Effectively, new methods were developed with the *verification* step after the *character generation* step, the *grouping* step and both.

Method	Segmentation	Detection	Verification
Phan et al. [90]	laplacian output	text/non-text clustering via k-means	heuristics
Shivakumara et al. [104, 106, 107]	wavelets	text/non-text clustering via k-means	heuristics
Shivakumara et al. [102, 103, 105]	multiple edge filters	block growing + recursive profiling	heuristics
Kim et al. [47]	local thresholding	heuristic based CC merging	heuristics
Liu et al. [60] & Jung et al. [42]	stroke filter output	morphological	SVM classifier
Dinh et al. [19]	locally adaptive edge detection	stroke width controlled dilation	heuristics
Epshtein et al. [24]	stroke width transform	heuristics	-

Table 2.2 Text detection methods from 2006-2010. The methods still focussed on caption text with image processing techniques. Major works based on components were done in latter part of this period using concepts like stroke width.

This decade also witnessed broadening of the scope of such methods,

1. Methods shifted from caption text in videos to scene text images [44, 82, 83, 85, 100], text tracking in videos [32, 72, 74], image retrieval via scene text [39, 76] as well as several integrated pipelines with detection and recognition combined [9, 40, 82, 116, 129].
2. Scene text was further classified into born digital text [97], accidental scene text and focussed scene text [45, 97].
3. Text orientation constraint was relaxed from horizontal to any orientation. Several datasets were released with text in any orientation like the MRRC [53] and the MSRA [129] datasets with text multiple languages like English, Korean, Chinese and Kannada.

Simultaneously, the Robust Reading Competition (RRC) was organized in 2011 [97], 2013 [45] which gained a lot of attention due to release of standard datasets as well as significant improvement in performance as compared to the older texture based methods. The ICDAR 2011 RRC winner was Kim's method which clustered MSERs heuristically into possible text blocks and then uses an adaboost classifier to reject false positives. The ICDAR 2013 RRC winning method by Yin et al. [132] proposed a similar method (Figure 2.3) but clustered MSERs using agglomerative clustering techniques and a superior adaboost classifier. Several other interesting works like scene text detection based on visual attention models [98] to T-HOG descriptor [75] for single line text detection were also proposed during this era. The emergence of deep learning methods via convolutional neural networks provided robust classification of characters and text patches [9, 38, 40]. Survey papers on emerging trends in scene text were also published recently [133, 136]. A summary of important methods in this time period is shown in Table 2.3.

Component based approaches Several component based approaches of this time period performed well on the standard datasets by relying on the MSER algorithm. [82] worked on the component tree and effectively pruned the MSER lattice by verifying the components in the MSER component tree. Real-time systems were developed which worked on the MSER component tree, proposing incrementally computable features [83] for the components in the tree for character detection. Stroke orientation based character candidate selection was proposed in [85] where a character was represented as a set of strokes. In this way a character classifier was developed which used a set of oriented filters. As a graph labelling problem, [44] used the higher order correlation clustering of MSERs to generate graph connections with MSERs as nodes. Weak hypotheses were generated by coarsely grouping MSERs based on spatial alignment and consistency. These served as cues for long range interactions while partitioning the MSERs into text line candidates. An energy minimization framework was proposed by Shi et al. [100] where a random forest classifier was trained to discriminate characters from non-characters. The scores provided the unary potential while similarity of the components constituted the pairwise potential.

The methods discussed above followed the framework where the classifier from the *verification* step was introduced early in the system. Yin et al. [128, 132] proposed a system where they shifted the classification task after grouping. Their method consisted of single linkage clustering via a distance metric learning approach to MSERs to group them into clusters. These clusters were then classified using a character classifier to estimate the posterior probability of the cluster to be text. Gomez et al. [30, 31] propose a similar system where they perform single linkage clustering on MSERs and use an adaboost classifier to reject non-text clusters. These works focussed on multilingual and multi-orientation text extraction and did not require any heuristics to reject non-text candidates. They presented a simple yet effective framework targeting a larger spectrum of text detection challenges.

Text region proposal methods A contrasting approach for detection also came into the picture in this time period. Focussing on the role of recall in the detection part of an end to end pipeline, the need for an explicit localization of text could be bypassed using a region proposal method. They were presented as an alternative to sliding window strategy for the end to end framework with lesser number of windows generated, relying on a recognition engine to improve the precision of the detection result. Treating text as objects, this approach followed the style of top performing PASCAL object detectors [36]. In the recent years, several generic object region proposal algorithms were proposed in the field of object recognition which can be considered for text region proposal. Super-pixel based methods like [114] were proposed, where the authors group the super-pixels hierarchically and use several diversification strategies comprising of complementary similarity measures to get the proposals. Super-pixels were also considered in a graph cut framework in [92] where a local search to merge adjacent super-pixels was performed followed by a global search using graph cut segmentations. Manen et al. [67] use the super-pixels connectivity graph to generate random partial spanning trees as proposals using the randomized version of prim's algorithm. A CRF based proposal generation method was proposed in [23]

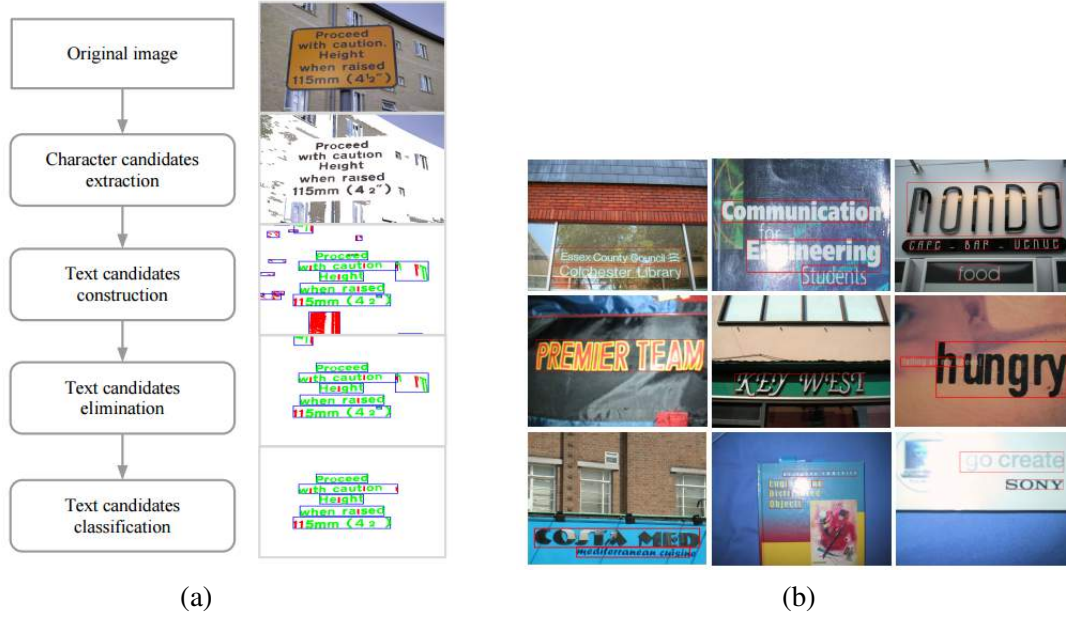


Figure 2.3 Overview of a robust text detection method by Yin et al. [132]. (a) shows the flowchart of the system with character candidates as MSERs. (b) contains a few qualitative results of the system on images from ICDAR 2011 dataset.

on super-pixels with diverse solutions on multiple scales followed by re-ranking them based on different appearance cues. Region proposal was also presented in a Bayesian framework in [5] which combined several cues like location, size, color contrast, edge density, etc. to extract set of superpixels which appear different from their surroundings and form a closed boundary. Multi-scale hierarchical segmentation with grouping strategies to combine multi-scale regions into object candidates by exploring combinatorial space was suggested in [7]. An edge based method was proposed in [137] where box objectness scores were computed based on the number of edges inside the box minus the number of contour members overlapping the box boundary. A scene text region proposal specific method was proposed by Jaderberg et al. [40] where a patch based text/non-text CNN classifier was used to generate proposals at various scales. In [39] another region proposal method targeting scene text was discussed which used an aggregated channel feature detector [22] and the edge boxes method [137] to generate complementary set of proposals resulting in high recall.

2.3 A flexible detection scheme

In this section, we propose a fusion of two different approaches to detection, one to explicitly detect text while the other to generate several text region proposals. The text detection system is based on Gomez et al. [31] which relies on hierarchical clustering of MSERs followed by a text/non-text classifier and emphasizes on achieving a better overall f-measure. The text region proposal method utilizes the

Method	Character Generation	Grouping	Verification	Remarks
Neumann et al. [82, 83]	MSERs	heuristics	SVM classifier for MSERs	real-time, end to end
Neumann et al. [85]	CCs via oriented stroke detection	heuristics	K-NN classifier for CCs	end to end
Kim et al. [97]	MSERs	heuristics	adaboost classifier for text blocks	RRC 2011 winner, real-time
Koo et al. [52]	MSERs	adaboost classifier based clustering	MLP classifier for text blocks	two classifier system
Shi et al. [100]	MSERs	heuristics	MSER labelling via graph partitioning	-
Yin et al. [128, 132]	MSERs	agglomerative clustering	adaboost classifier for text blocks	RRC 2013 winner
Gomez et al. [30, 31]	MSERs	agglomerative clustering	adaboost classifier for text blocks	multi-orientation and multi-script

Table 2.3 Text detection methods from 2011 to present. Methods in this era relied on vision techniques like MSERs, graph cuts, sliding window based approaches, etc. and relied more on supervised learning via CNN, adaboost, SVMs for text or character classification during the verification step.

patch based text/non-text CNN classifier from [40] to generate text region candidates with high text recall. In this section, we first discuss the importance of precision and recall in detection systems, justifying the need for a high recall detection scheme. It is followed by the description of the text detection system and the text region proposal method with their various components. In the end, we fuse the two methods, utilizing the proposals to improve the detection results. A flowchart of the detection pipeline is shown in Figure 2.4.

2.3.1 Precision or Recall: What is important in detection

Text detection systems are generally evaluated in terms of precision, i.e., percentage of correct text regions over all detected regions, and recall, i.e., percentage of all correctly detected regions over all correct regions. Their harmonic mean i.e. the f-measure is also considered as a single metric to evaluate performance. From the ICDAR RRC 2011 [97] and 2013 [45], we observe that the top performing methods have a precision 15-20% more the recall even though the f-measures of the methods have increased over time. This, however may not be an optimal design choice, especially when our goal is to build an end to end recognition framework. A high precision detection system can perform no better than a high recall detection system when clubbed with the same recognition engine, assuming we can use the recognition scores to prune the detection results. Since detection systems generally employ a

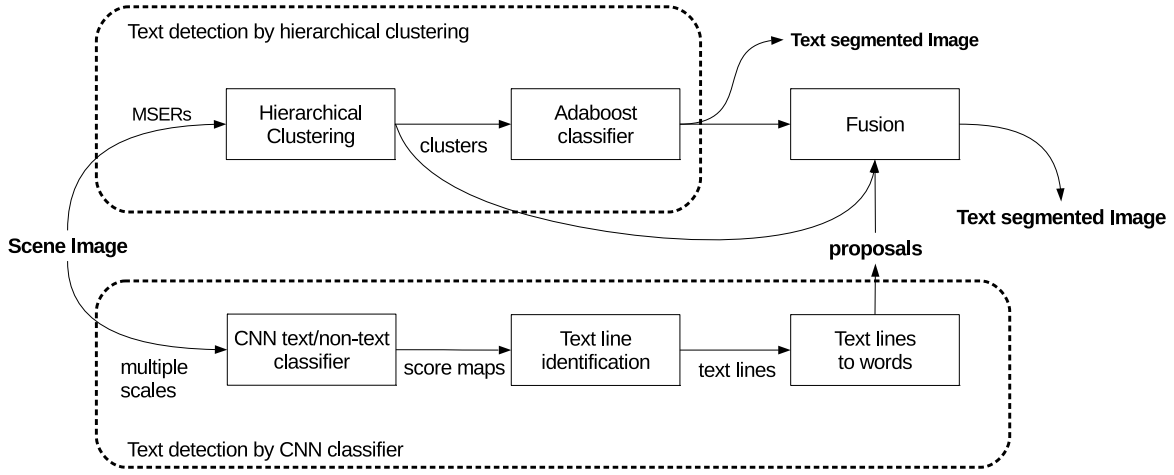


Figure 2.4 Text detection pipeline comprising of a text detection method and a text region proposal method. The text detection method classifies the MSER clusters present in the input image to generate a binarized image with text as foreground. The proposal method utilizes a CNN classifier to generate a score map which is used to discover probable text lines and words on the image. These two methods with the complementary nature of attaining high precision and high recall are fused to get a flexible detection scheme.

simple text/non-text classifier as compared to the more expensive multi-class character classifier in a recognizer, the detection recall can facilitate the better overall performance of an end to end framework.

2.3.2 Text Detection via Hierarchical Clustering

The system relies on two major components (i) an effective hierarchical clustering method which agglomerative groups components based on similarity and, (ii) a classifier designed to classify such clusters as a whole as text or non-text. The system begins by finding the MSERs in the image which have been demonstrated to perform consistently in the task of grouping pixels of characters into a single component. Besides that they can be extracted from an image in real time as a set of components with each component being defined as a set of pixels. This allows us to compute simple and effective features in real-time. Since text lines have characters with similar features like colors, stroke width or sizes, they can be effectively used to cluster characters into groups.

Effective hierarchical clustering Let R_n be the set of n regions detected by the MSER algorithm on the given image. We perform an agglomerative clustering procedure via the single linkage method where initially each region $r \in R_n$ is considered as a cluster itself and at each step two of the closest clusters (A, B) are merged together using the criterion $\min\{d(r_a, r_b) | r_a \in A, r_b \in B\}$ until all regions are grouped in one cluster giving us $n - 1$ clusters. The distance metric d is calculated as a weighted sum of various features extracted from each region namely,

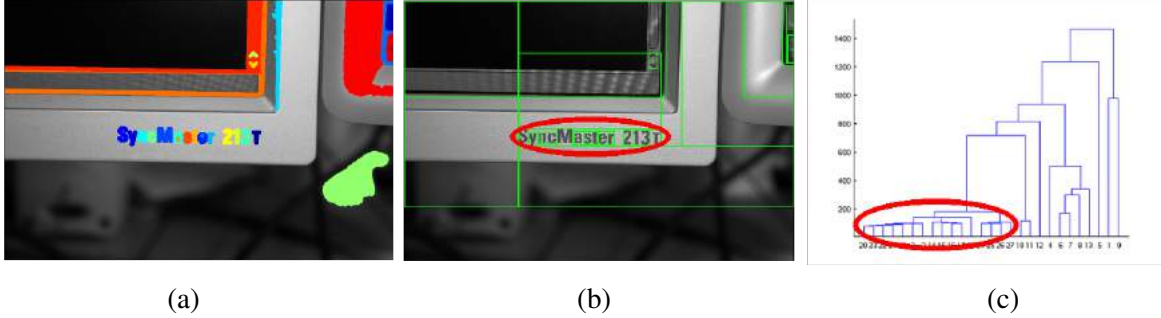


Figure 2.5 Single linkage clustering on MSERs from a typical scene image (a) with clusterings in (b). We observe that characters tend to group with the adjacent characters, hence creating a word level cluster with low intra-cluster distance as shown in the dendrogram (c). The text in red circle annotation and its corresponding nodes in the dendrogram suggest that characters from a text line group with each other first.

- mean gray value of the region
- mean gray value of the immediate outer boundary of the region
- region's major axis length
- mean stroke width
- mean of the gradient magnitude at the region's border

This 5 dimensional feature space is combined with region's centroid co-ordinates as a spatial constraint during the clustering process. The color features are normalized by maximum intensity while the region's major axis and centroids are normalized by the image size. Given two regions (r_a, r_b) with their features (a, b) and centroids, the distance function is defined as,

$$d(a, b) = \sum_{i=1}^5 w_i (a_i - b_i)^2 + (x_a - x_b)^2 + (y_a - y_b)^2 \quad (2.1)$$

The single linkage clustering process results in a dendrogram which represents the order of connections amongst the regions. Figure 2.5 shows the clusters on a sample scene image with words in the initial clusters. The euclidean distance metric d ensures rotation invariance within the regions i.e. the order of connections remains the same as the image is rotated. This approach allows grouping of characters irrespective of their orientation.

The weights of individual features were found out by performing a grid search over all possible combinations of weights. The search procedure aims to find the weight configuration which gives the maximum number of pure text groupings (i.e., clusters with characters only) over a dataset. In [31], the grid search is performed on the ICDAR 2003 and MRRC training set containing 229 and 167 images respectively. The optimal weights obtained were $w_{opt} = (0.65, 0.65, 0.49, 0.67, 0.91)$ for the five features.

Text/non-text classifier After clustering, we get a set $n - 1$ clusters from the R_n regions. The regions can now be classified as text or non-text and all text classifiers can be merged to segment text from the image assuming all components in the text clusters are characters. For each cluster comprising of a set of regions, we build a 2-D minimum spanning tree(MST) using their region centres and calculate the following cluster level features,

- foreground intensities standard deviation
- background intensities standard deviation
- major axis coefficient of variation
- stroke width coefficient of variation
- mean gradient standard deviation
- aspect ratio's coefficient of variation
- average euclidean distance of the seven Hu's invariant moments
- convex hull compactness mean and standard deviation
- convexity defects coefficient of variation
- MST between-node angles mean and standard deviation
- MST edge widths coefficient of variation
- MST edge distances mean and region diameters mean ratio

These features can be quickly computed by using the already computed MSER features. Building the MST is a one time operation and its corresponding features are quickly computable. The coefficient of variation is calculated as the ratio of standard deviation to the mean.

Using these features, we train an adaboost classifier using the training set of ICDAR and MRRC datasets. We obtain two sources of data from each dataset in the form of, (i) ground truth pixel wise segmentation giving us word level groups which can be assumed as the output of a clustering step and, (ii) finding MSERs and clustering them using w_{opt} parameters and considering clusters with ground truth overlap of more than 80%. This gives us around 3K positive samples and 25K negative samples and we train an adaboost classifier by down-sampling the negative samples to 3K. After training, we run the classifier on the remaining 22K negative samples to find top 1000 false positives. We use them to retrain the system again which improves the performance of the classifier. Upon parameter search, we find an adaboost classifier with 100 trees of depth 3 to perform the best.

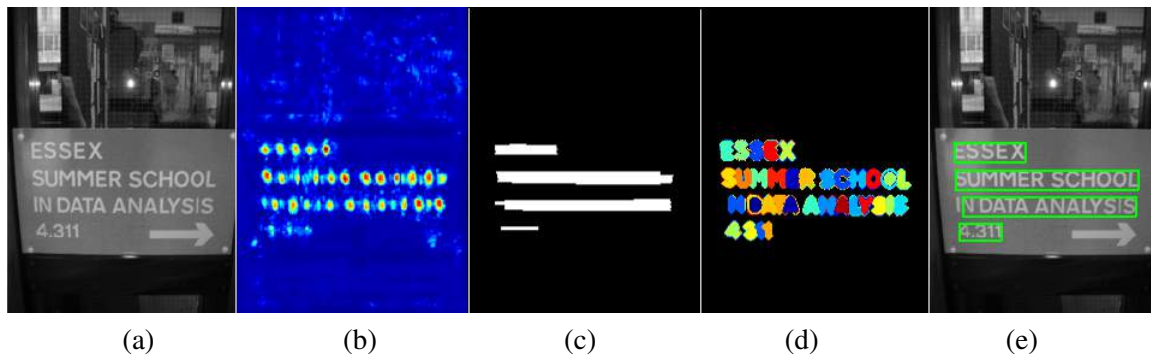


Figure 2.6 Text region proposal on a scene image (a) at a specific scale. (b) shows the score map using the patch-wise CNN classifier with higher scores in text regions, (c) gives us the text lines found after thresholding the score map and applying RLSA, (d) shows us the components overlapping the text lines and, (e) shows the final proposed text regions in bounding boxes.

2.3.3 Text Detection via CNN Classifier

The text region proposal system tries to generate several word level text regions with maximum recall. Their objective is to provide a very small set of regions on the image with high recall as compared to the set of regions generated using sliding window technique. It thus enables the scope of a feedback loop to the actual detection of text by considering the quality of recognition for each text region proposed. In this system we use a 4 layer CNN classifier trained to distinguish text patches from [40] which gives a classification accuracy of 98% and 97% on ICDAR 2003 and SVT datasets. Given an image at a single scale, the classifier is used according to the following procedure (Figure 2.6) to generate proposals,

1. **Text saliency map generation** - The process begins by running the text/non-text classifier patch wise thus giving us a score map as an output. The classifier responds with high scores to patches where text is present in a similar scale. The score map thus ideally has high scores in the middle of characters which gradually decrease as we go to the boundaries of the character.
2. **Text line identification** - The score map is then binarized using the Run Length Smoothing Algorithm (RLSA) technique which is a row wise operation joining possible adjacent character regions together. The technique is presented in algorithm 1 and used with parameters $t_1 = 2, t_2 = 3, t_3 = 0.5$. The output binarized image has possible text lines in foreground.
3. **Text line to words** - Each text line is now used to find a set of connected components from the image via the MSER method which significantly overlap with it. Using these connected components from all text lines as foreground, we get another binary image. Adjacent components are now connected again using the RLSA algorithm to get word level bounding boxes.

Algorithm 1: Run Length Smoothing Algorithm (RLSA)

input : Score Map S , Thresholds t_1, t_2, t_3
output: Binary Image I
Thresholded Score Map $I = \text{threshold}(S, t_1 * \text{mean}(S))$;
for each row R **in** I **do**
 $\text{SpacingList} \leftarrow \text{lengthBlankSpaces}(R)$;
 $T = t_2 * \text{mean}(\text{SpacingList}) + t_3 * \text{stdDev}(\text{SpacingList})$;
 for each Space p **in** SpacingList **do**
 if $p \leq T$ **then**
 $R \leftarrow \text{fillSpace}(R, p)$;

The word level bounding boxes are generated for 16 different scales of the image targeting text heights from 16 to 260 pixels. They are filtered based on geometric constraints like height and aspect ratio. The bounding boxes also go a non-maximal suppression after sorting them by their average text saliency score from the score map.

2.3.4 Fusion

In this section, we propose a fusion of the two text detection methods discussed in this section. We use the proposals as a simple classifier running in parallel with the adaboost classifier of the detection by hierarchical clustering method. Leveraging on the high recall of the proposal method, we classify the clusters as text if both the following criteria are met,

1. Overlap of the cluster with one of the proposals is above a threshold T
2. The maximum overlap proposal has a mean CNN score above a threshold S

The first criterion ensures that the cluster must be one of the proposals with a carefully tuned threshold T . A low threshold T would relax the restriction, allowing additional components in the cluster to be considered as well which in turn, will reduce the precision. A high threshold T would lead to no matches, as the methods are independently producing detection results. The second criterion ensures that the chosen proposal, i.e. the one with the maximum overlap above the threshold T , has a high probability of containing text.

2.4 Experimental Analysis

In this section, we mainly evaluate the implemented methods on popular datasets. We compare them with methods whose code is publicly available either from the original authors or implemented in a library.

Dataset Name	Train Images	Test Images	Language	Orientation	Ground truth Type
ICDAR 2003 [64]	250	250	English	Horizontal	Boxes + Segmentation
ICDAR 2011 [97]	229	255	English	Horizontal	Boxes Only
ICDAR 2013 [45]	232	236	English	Horizontal	Boxes Only
MRRC [53]	167	167	English + Kannada	Any	Boxes + Segmentation

Table 2.4 Various text detection datasets used by the community to benchmark the performance of the methods. The MRRC dataset was introduced in a competition at ICDAR 2013 for multi-orientation multi-script text detection. The ICDAR 2003 dataset has pixel level ground truth shared by the authors of [73].

2.4.1 Datasets

For evaluation purposes, we rely mainly on the ICDAR Robust Reading Competition Datasets. A complete description of the datasets used are present in Table 2.4.

2.4.2 Evaluation

Given an image I with a set of ground truth boxes B_{gt} with N boxes and ground truth binarized image I_{gt} , these are the evaluation metrics for the following methods,

- Text detection - Given the detection result as B_{det} with M boxes, we can calculate precision and recall to measure the performance of such systems as per the following protocols,

- ICDAR 2003 [64] - The $overlap(b, b')$ is measured as intersection upon union of the bounding boxes b and b' . F measure is computed as the harmonic mean of precision and recall.

$$precision = \frac{\sum_{b \in B_{gt}} m(b, B_{det})}{\|B_{gt}\|} \quad (2.2)$$

$$recall = \frac{\sum_{b \in B_{det}} m(b, B_{gt})}{\|B_{det}\|} \quad (2.3)$$

$$m(b, B) = \max_{b' \in B} overlap(b, b') \quad (2.4)$$

- ICDAR 2011 [97] - This evaluation uses the framework proposed by Wolfe et al. [124] which takes into account the overall quality of the matches between ground truth and detections. Matches are first determined using area overlaps, given certain quality thresholds. Thereafter, different weights for one-to-one, many-to-one and one-to-many matches

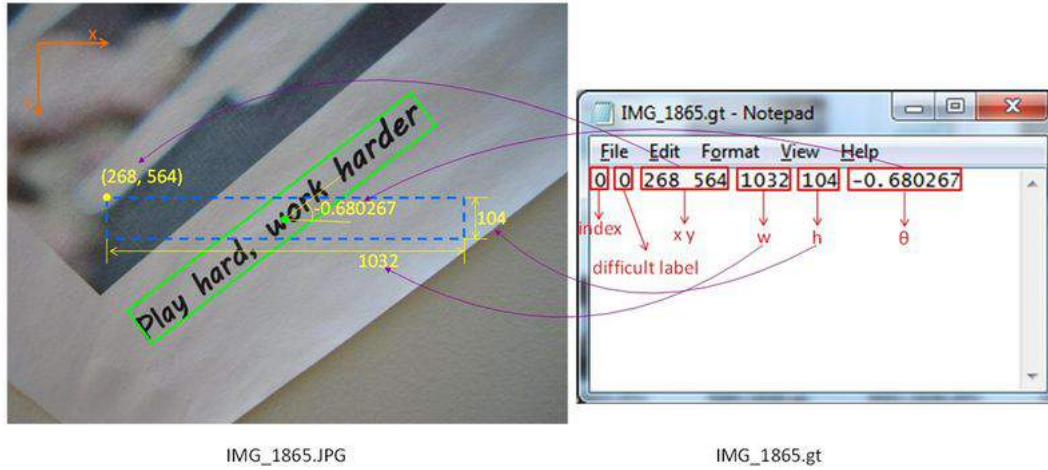


Figure 2.7 Ground truth format illustration for multi-orientation scene text detection for the MSRA dataset. The ground truth is available as line level boxes, thus bypassing the word level splitting task which is mostly heuristic based. It also reduces the chances of over-segmenting a word, which might be undesirable for the recognizer. Typically can obtain a minimum bounding rectangle for a set of points using functions such as *minAreaRect* in OpenCV.

are used to pool the results. The default thresholds are used i.e. area precision $t_p = 0.8$ and area recall $t_r = 0.4$. For many-to-one matches no penalty is assigned while one-to-many matches are weighted by 0.8. The evaluation algorithm is implemented in free to use DetEval software¹.

- MSRA 2012 [130] - This evaluation protocol is used for text in arbitrary orientation. The ground truth consists of rotated bounding boxes with the box location, width, height and orientation as shown in Figure 2.7. Given a detected bounding box with a specific rotation and its corresponding ground truth, the boxes are rotated to zero degrees and their area overlap is checked. If the orientation difference is less than $\pi/8$ and overlap ratio is more than 0.5, it is considered as a correct match. The ground truth is present at text line level and the system is evaluated on the usual precision and recall statistics.
- Text segmentation - Given the segmentation result as a binary image I_{det} , we can calculate precision and recall to compare methods. We denote the number of pixels labelled as foreground or text in a binary image using $\|I\|$. There are two main protocols to evaluate such results,

¹<http://liris.cnrs.fr/christian.wolf/software/deteval/>

- ICDAR 2003 (Pixel level) [64] - We calculate the precision and recall in terms of overlap of the detected and ground truth pixel information,

$$precision = \frac{\|I_{det} \cap I_{gt}\|}{\|I_{det}\|} \quad (2.5)$$

$$recall = \frac{\|I_{det} \cap I_{gt}\|}{\|I_{gt}\|} \quad (2.6)$$

- ICDAR 2011 (Atom level) [97] - This evaluation protocol uses the framework by Clavelli et al. [16] where atoms or smallest combination of text parts are defined. The quality of the segmentation is measured by the degree to which the detections are able to preserve the morphological properties of ground truth instead of simple counting of mislabelled pixels. In simple terms, the method allows slight dilation and erosion of the ground truth but penalizes detected atoms which are distinctly different shapes. To consider a match between a detection atom and a ground truth atom, two conditions must be fulfilled, (i) minimum coverage condition, satisfied if detected atom covers a minimum percentage of pixels (controlled by parameter T_{min}) in ground truth atom and, (ii) maximum coverage condition, satisfied if no pixel in detected atom lies outside the maximal area (defined by parameter T_{max} controlling the dilation as a function of stroke width of the component) of ground truth atom. For standard evaluations, we set the parameters as $T_{min} = 0.5$ and $T_{max} = 0.9$. Each atom in the ground truth is now classified as well segmented, merged, broken, broken and merged or lost while false positive responses are also counted. Atom level precision, recall and f-measure are calculated over the whole collection of detected atoms.
- Text region proposal - Such methods are evaluated using the recall measure. Generally, recall is computed as in 2.3 while the maximum overlap is thresholded at a level T . Besides recall, we also calculate the average number of windows per image and the average time taken per image.

2.4.3 Text Detection via Hierarchical Clustering

In this section we perform experiments on the standard datasets ICDAR 2003 and 2011 and evaluate results for detection via segmentation and bounding boxes. We compare the implemented method with other recent methods whose source code is open and publicly available. A comparison of the verification step classifiers for all the methods including the implemented one is present in Table 2.5.

For the implemented method, we use the parameters $\delta = 19$, $maxvariation = 1$, $mindiversity = 0.7$ to generate MSERs on gray images. We allow MSERs with area less than 0.05% and more than 0.00002% of the whole image area. We also reject clusters of size more than 50 as they are less likely to contain text. We measure the detection and segmentation results of the method with single linkage clustering + adaboost classifier (SLC+adaboost) on the MRRC and ICDAR2003 datasets and compare it with other methods in Table 2.6. Few successful qualitative results are shown in Figure 2.9.

Method & Source	Features	Verification classifier information
Gomez et al. [30] (Github)	Handcrafted	Adaboost trained on ICDAR 2003 and MSRA 2012
Milyaev et al. [73] (Web)	None	Pixel-wise segmentation via graph cuts
CSERs [83] (Github)	Incrementally computable descriptors	Adaboost trained using ICDAR 2003 and 2011
Yi et al. [55] (Github)	Stroke Width + HOG@edges + Perceptual Divergence statistics	Naive Bayes model trained on ICDAR 2013
SWT [24] (Github)	Stroke Width based features	Heuristics using thresholds found empirically
Our Method	Handcrafted	Adaboost trained using MRRC, ICDAR 2003 and 2011

Table 2.5 List of text detection implementations used for evaluation along with their source code links. The verification step involving classifying characters or group of characters as text or non-text is summarized. Features used by the classifier, the classifier type and their training datasets are mentioned in this overview.

We observe that SLC + adaboost method outperforms its very similar method by Gomez et al. [30] on the MRRC dataset and obtains comparable results on the ICDAR 2003 dataset. Both the methods share the same hierarchical clustering mechanism to group MSERs but the former uses a single set of clustering with optimal weight for each feature from the MSERs. The latter uses an evidence accumulation technique using multiple clusterings arising from each MSER feature. Also our implementation is trained on a better feature set with more training data resulting in an overall improvement in segmentation results. We plot the precision recall curves in Figure 2.8(a) for both these methods by varying the classifier acceptance threshold. While the SLC + adaboost contains a single adaboost classifier, the method by Gomez et al. [30] has two classifiers, one used for the several clusters from multiple clustering strategies and the second for the single set of clusters after evidence accumulation. The pixel-wise segmentation method by Milyaev et al. [73] outputs two binarized images and the one with higher f-measure is retained for evaluation. Due to the absence of a classifier based verification step and components based character extraction and reliance on image processing techniques to generate unary and pairwise terms, the method does not perform well.

We also evaluate the SLC + adaboost detection results via bounding boxes with the help of Tesseract OCR [109]. We run the OCR on the pixel wise segmented output image and get word level bound-

ing boxes. The recognition result is discarded and not utilized for improving the detection result. The detected word bounding boxes are then evaluated against the ground truth boxes on the ICDAR 2003 and 2011 datasets using their corresponding metrics. The Stroke width implementation performs the best even though it relies on heuristics in its verification step. Its success is mainly attributed to the powerful assumption that text has a uniform stroke width. Our implementation outperforms the faster CSER implementation mainly due to a cluster level text/non-text classifier as compared to a character classifier. We plot the precision recall curves for the SLC + adaboost and the CSER method in Figure 2.8(b) by varying the classifier acceptance thresholds while keeping the MSER parameters fixed to the default values. We observe that the CSER method generates a tighter PR curve despite of changing the thresholds of both the classifiers involved while the SLC + adaboost method loses precision from 70 to 45 but gains recall from 5 to 55.

Performance Issues Success of any method is generally dependent on the quality of text present in the scene image. Images with text comprising of several words in simple colors and layout can be effectively detected with high recall due to, (i) the ability of MSER method to successfully extract characters as components (observe that the PR curves in Figure 2.8 saturate at a recall around 60) and, (ii) presence of characters in a group which strengthens our confidence of those particular components to be text. Images with very less text, say only one or two characters, as well as with complex colors and layout can be tougher to detect as the MSER method may fail to select the characters as a single component either by, (i) over-splitting in case of fancy fonts or, (ii) under-splitting due to very less character spacings. Moreover, if the text possesses less contrast between its foreground and the background, the MSER parameters are susceptible to miss the characters. Overall, if we relax the MSER parameters and allow them to return several character candidates, our character recall increases but adds several false positives. The precision of any method is however, is governed by, (i) the non-text content in the scene image and, (ii) strength of the classifier at the *verification* step. Ideally the precision and recall trade-off can be evaluated by the MSER parameters and classifier scores, but it is more sensible to consider the MSER parameters as a design choice, governed by the type of scene images the algorithm is intended to process.

2.4.4 Text Detection via CNN Classifier

For region proposal methods, we evaluate the method using recall with $T = 0.5$, average number of proposals and runtime. We use the implementations of other object region proposal methods² for comparison purposes. We show the results on ICDAR 2003 and 2013 datasets in Table 2.7.

We observe that the edge boxes method by Zitnick et al. [137] gives us the highest recall of 83.7, but at the same time generates around 6K proposals. On the contrary, the CNN based method has a recall of 75 only but greatly reduces the number of proposals to around 1.5K. The superpixel based methods [5, 23, 67, 92] do not perform well as compared the edge boxes method except for hierarchical grouping of superpixels strategy by Uijlings et al. [114] which is able to generate proposals as clustering

²<https://github.com/batra-mlp-lab/object-proposals>

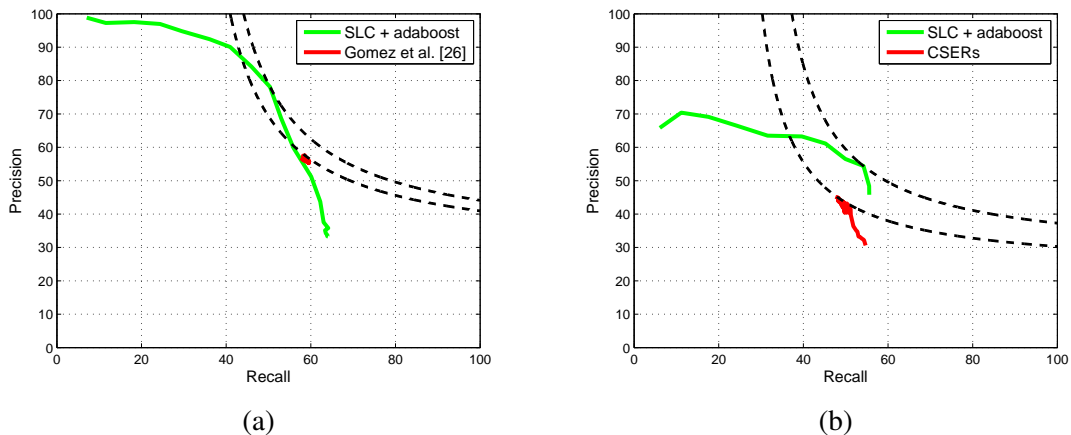


Figure 2.8 Precision recall curves for the classifier based methods on the ICDAR 2003 dataset. (a) shows the detection by segmentation PR curve (pixel level evaluation) for the SLC + adaboost method and Gomez et al. [30]. The methods achieve an f-measure of 59 and 58. (b) shows the detection by bounding boxes PR curve (ICDAR 2003 protocol) for the SLC + adaboost and the CSER method with best f-measures of 54 and 46 respectively. The best f-measures for the respective methods in both the graphs are shown as dashed lines.



Figure 2.9 Qualitative results for the SLC + adaboost method on images from ICDAR 2003 dataset with original images in the first row and their corresponding segmentation results in the second row. The method is able to achieve near optimal results even in the presence of repeating (e.g., bricks, table textures, etc.) patterns and non-text components with the same stroke color to that of the text regions.

Method	MRRC			ICDAR 2003			Runtime
	Recall	Precision	F-measure	Recall	Precision	F-measure	
Gomez et al. [30]	60	67	63	59	58	58	3
Milyaev et al. [73]	82	20	32	74	31	43	7
SLC + adaboost	60	78	69	53	66	59	10

(a) Text detection via segmentation

Method	ICDAR 2003			ICDAR 2011			Runtime
	Recall	Precision	F-measure	Recall	Precision	F-measure	
CSERs [83]	46	46	46	46	42	44	3
Yi et al. [55]	48	58	53	61	48	54	4
SWT [24]	62	65	60	59	61	60	6
SLC + adaboost	57	49	54	48	42	45	12

(b) Text detection via bounding boxes

Table 2.6 Text detection results of various methods on the popular datasets with average runtime in seconds. The segmentation results are evaluated at pixel level while the bounding boxes are evaluated using their respective protocols. The segmentation methods are implementations shared by the original authors while the CSERs and SWT methods in detection section are implementations available in OpenCV 3.0 contrib module and CCV library. Detection evaluation for SLC + adaboost method is done using word level bounding boxes returned by Tesseract.

text superpixels together. Overall, due to the missing recognition module output as well as the lack of one single measure combining recall, proposals per image and time taken, it becomes difficult to distinguish between similar performing methods.

Re-ranking with a recognizer Additionally, we also evaluate the proposals using Tesseract OCR and rank them based on the OCR recognition scores. The proposals with overlap more than 50% with any ground truth are labelled as positive. We plot the precision recall curves for some of the high recall methods from Table 2.7 in Figure 2.10. We observe that CNN classifier based text proposal method has a higher overall precision, suggesting that the correct proposals are easier for the OCR to recognize. In case of the proposals from other methods, the correct proposals may not be regions with pure text which leads to a poor OCR score.

2.4.5 Fusion

We evaluate the segmentation results of the fusion on the ICDAR 2003 dataset. We choose the parameters $T = 0.9$ ensuring that fairly similar cluster bounding boxes are only selected. We set

Method	ICDAR 2003			ICDAR 2013		
	Recall	R/I	Time(sec.)	Recall	R/I	Time(sec.)
Endres et al. [23]	66.0	1417	1216	64.0	1510	1300
Arbelaez et al. [7]	69.5	1096	75	69.0	1109	137
Alexe et al. [5]	58.0	1000	9	56.0	1000	10
Manen et al. [67]	72.7	1144	4	70.3	1208	8
Rantalankila et al. [92]	26.7	676	102	22.8	663	662
Uijlings et al. [114]	79.0	4168	13	72.1	4053	15
Zitnick et al. [137]	83.7	6270	3	79.2	6409	7
CNN based	75.0	1658	10	72.0	1400	11

Table 2.7 Text region proposal results of various methods on ICDAR datasets. The CNN based region proposal method obtains a fairly high recall as compared to the method with best recall but at the same time has lesser number of region proposals per image (R/I).

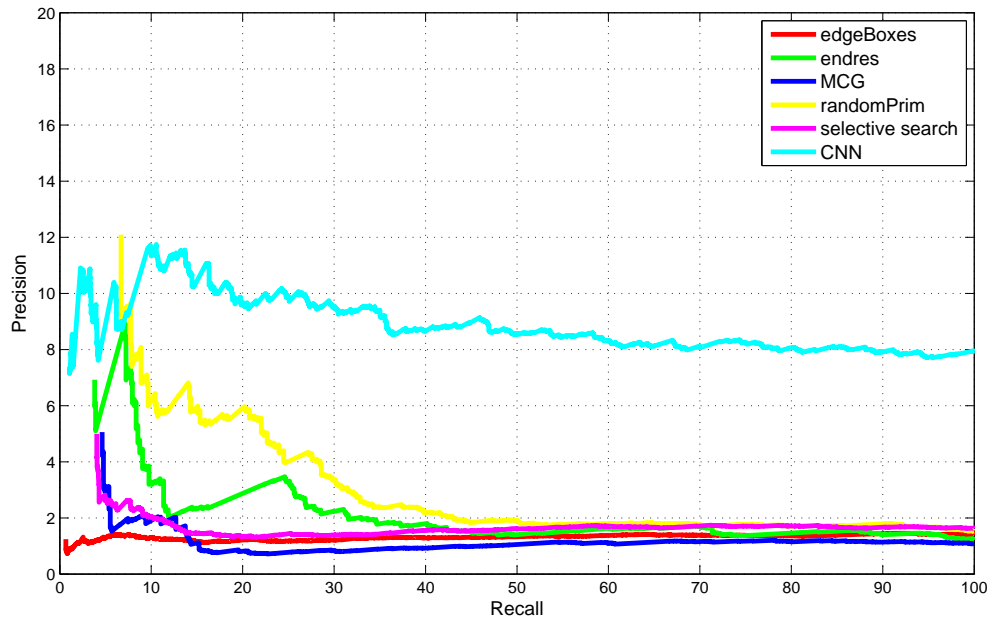
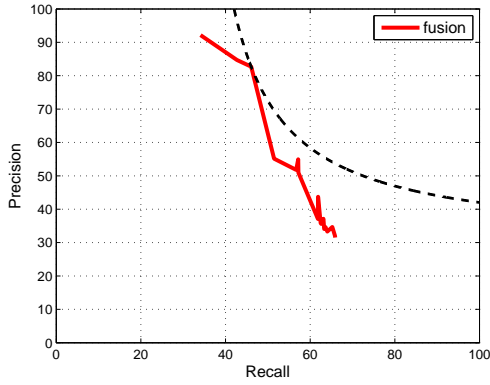


Figure 2.10 Precision recall curve for various proposal methods. We observe that all the methods start with a good precision suggesting actual text proposals being ranked higher but lose precision as all the proposals are taken into account.

the CNN score threshold as $S = 5.0$ through cross-validation. We plot the precision recall curve in Figure 2.11(a) by varying the adaboost acceptance threshold and the parameter S . In Table 2.11(b), we observe that we get a better f-measure than the previous SLC + adaboost scores at a cost of lower recall but higher precision. Specifically, the operating point with the best f-measure for the fusion has the SLC + adaboost performance with recall of 42 and precision of 86. Hence, a high precision detection by the SLC + adaboost method is fused with a complementary high recall text proposal method to get an overall gain in f-measure.



(a)

Method	Recall	Precision	F-measure
Gomez et al. [30]	59	58	58
SLC + adaboost	53	66	59
Fusion	46	83	60

(b)

Figure 2.11 Text detection performance of the fusion on ICDAR 2003 dataset. (a) shows the precision and recall curve by varying the adaboost classifier threshold and parameter S . Curves with constant f-measure are shown in dashed lines. Table (b) compares the fusion performance with the other methods.

2.5 Summary

We provide a survey of important works in scene text detection from the last fifteen years. We categorized them based on the techniques used and how they evolved over time. We discussed a text detection method which uses hierarchical clustering on connected components. Clusters labelled as text by a classifier are retained, thus giving a segmentation of text from its background. A text region proposal method is also discussed which uses a patch based CNN classifier for text/non-text classification. Several proposals as bounding boxes are generated with high text recall. Both the complementary methods are fused to obtain better performance. The methods are evaluated on standard datasets and compared with other publicly available methods.

Chapter 3

Scene Text Recognition and Retrieval

3.1 Introduction

Text can play an important role in understanding street view images. In light of this, many attempts have been made to recognize scene text [9, 29, 78, 94, 117, 120]. Scene text recognition is a challenging problem and its recent success is mostly limited to the *small lexicon setting*, where an image-specific lexicon containing the ground truth word is provided. Typically, these lexicons contain only 50 words [117]. This setting has many practical applications, but it does not scale well. As an example consider the scenario of assisting visually-impaired people in finding books by their titles in a library. Here the lexicon is populated with all the book titles. In this case, the small lexicon setting becomes less accurate as the lexicon sizes can range from a few thousands to a million. For instance, when lexicon size increases from 50 to 1000, the recognition accuracy drops by more than 10% [77, 94]. In other words, the general problem of scene text recognition, i.e., recognition with the help of a large lexicon (say a million dictionary words) is far from being solved. In this chapter, we investigate this problem.

One way to address the task of recognizing scene text is to pose the problem in conditional random field (CRF) framework and obtain the maximum a posteriori (MAP) solution as proposed in [77, 78, 86, 101, 117, 121]. In these frameworks, an energy function consisting of unary and pairwise potentials is defined, and the minimum of this function corresponds to the text contained in the word image. These methods demonstrated successful results in a small lexicon setting primarily due to the fact that the pairwise terms are computed with this lexicon have a positive bias towards the ground truth word. However, when the pairwise terms are computed from large lexicons, they become too generic, and often in such cases the MAP solution does not correspond to the ground truth. Besides this, MAP solutions suffer from drawbacks, such as (i) approximation errors in inference, (ii) poor precision/recall for character detection, (iii) weak unary and pairwise potentials. Consider the word “PITT” shown in Figure 3.1 as an example. The MAP solution for the word is “PITA”, which is incorrect. Our approach addresses this problem by using the top-M solutions to ultimately find text that is most likely contained in the image.

Word Image	Top-5 diverse solutions (ranked)
	PITA, PASP, ENEP, PITT , AWAP
	AUM, NIM, COM , MUA, PLL
	MINSTER, MINSHER, GRINNER, MINISTR, MONSTER
	BRKE, BNKE, BIKE , BAKE, BOKE
	TOLS, TARS, THIS , TOHE, TALP

Figure 3.1 Examples where the MAP solution is incorrect whereas the diverse solutions contains correct result as the pairwise priors become too generic when computed from large lexicons.

We begin by generating a set of candidate words with M-best diverse solutions [8]. With these potential solutions, we refine the large lexicon by removing words from it with a large edit distance to any of the candidates, and then recompute the M-best diverse solutions. These two steps are repeated a few times, which ultimately results in set of words most likely to represent the word contained in the image. Then a desired solution can be picked using various means (e.g., using minimum edit distance based correction using a lexicon). We show significant performance gain for recognition tasks in the large lexicon setting using this framework. We also present an application of computing the top-M solutions, i.e., text to image retrieval, where the goal is to retrieve all the occurrences of the query text from a database of word images. We will show that our strategy of re-ranking the words with the refined lexicon improves the performance over baseline methods.

3.2 Related Work

The problem of cropped word recognition has been looked at in two broad settings: with an image-specific lexicon [29, 78, 94, 101, 117] and without the help of lexicon [77, 120, 121]. Lexicon based recognition amounted to finding the best match from the lexicon to the word image. In [78], a CRF based solution was explored where the word image was segmented to find characters which represented the nodes. Using the bigram occurrence information as pairwise terms, the best label was inferred and the closest one from the lexicon was picked as the final solution. A holistic approach was explored by Jose et al. [94] where the lexicon and image features were embedded in a common space. Another holistic approach was proposed by [29] where the lexicon words were rendered and compared to the word image in image space via dynamic time warping. Shi et al. [101] proposed a part based tree structured model for character identification but required detailed annotated datasets for training. The lexicon free methods are relatively poor performing but desirable as they do not require a lexicon and are capable of detecting out of vocabulary words. Overall, the approaches for scene text recognition typically follow a two-step process (i) A set of potential character locations are detected either by binarization [9, 120] or sliding windows [78, 117], (ii) Inference on CRF model [77, 78], semi Markov model [120, 121], finite automata [86] or beam search [9] in a graph (representing the character locations and their neighbourhood relations) is performed. The PhotoOCR [9] presents a robust solution utilizing

multiple binarization techniques while intentionally oversplitting detected text lines and subsequently merging the splits to arrive at character segmentations with high accuracy. These approaches work well especially in small lexicon settings, but suffer from two main drawbacks: (i) obtaining a single set of true character windows in a word image in these methods is difficult as a single segmentation method may not be a robust choice, (ii) pairwise information gets less influential as the lexicon size increases. We adopt a similar framework in this chapter, but propose crucial changes to overcome the issues of previous approaches. First, we generate multiple word hypotheses and derive a set of candidate words likely to represent the word image. Second, we present a technique to prune the large lexicon based on edit distances between the candidate solutions and lexicon words. This proposed method allows us to significantly reduce the lexicon size and make the priors more specific to the image. Third, unlike prior works which yield a single solution, our method is also capable of yielding multiple solutions, and is applicable to the text-to-image retrieval task.

The remainder of the chapter is organized as follows. In Section 3.3.1, we present CRF framework for word recognition. We utilize multiple segmentations of word images to obtain potential character locations in section 3.3.2. We then present details of the inference method in section 3.3.3. Our lexicon reduction and pairwise term update steps are described in section 3.3.4. The two problem settings, i.e., recognition and retrieval, are then discussed section 3.4. Section 3.5 describes the experiments and shows results on public datasets. Implementation details are also provided in this section. We then make concluding remarks in section 3.6.

3.3 Proposed Method

We model the scene text recognition task as an inference problem on a CRF model, similar to [78], where unary potentials are computed from character classification scores and pairwise potentials from the lexicons. Small lexicon based pairwise potentials often help to recover from the errors made by character classification [99, 113]. However, when the pairwise potentials are computed from large lexicons, they become too generic, and the overall model cannot cope with erroneous unary potentials. To overcome this issue, starting from a large lexicon recognition problem, we automatically refine the problem statement and convert it to a small lexicon inference task.

The framework has the following components, as shown in Fig. 3.2: (i) Candidate word generation module, where we generate multiple words with each word as a set of characters spanning over the image, (ii) CRF inference module, where each word is represented as a CRF and inferred to obtain diverse solutions, and (iii) Lexicon reduction module, where we prune the lexicon by removing distant words after re-ranking the lexicon with a novel group edit distance computed using the diverse solutions. It is accompanied by re-computation of pairwise potentials which become image specific as the lexicon size decreases. We use different stopping criteria for recognition and retrieval tasks as we alternatively reduce our lexicon and infer solutions.

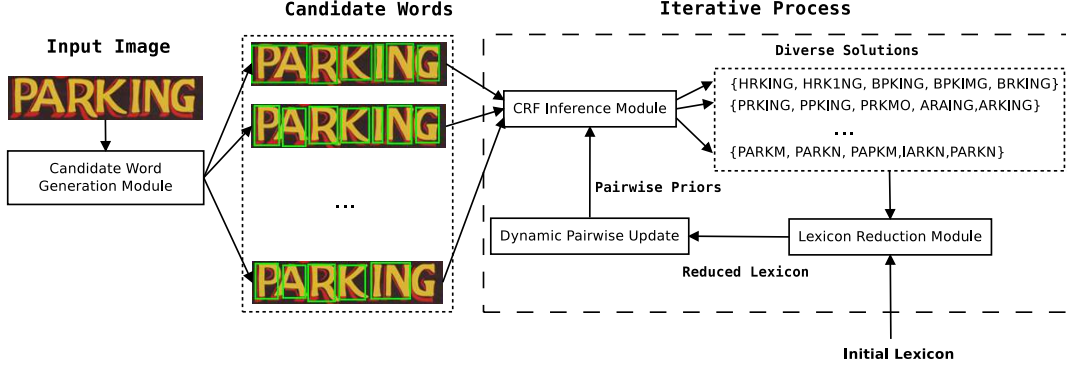


Figure 3.2 Overview of the proposed framework. The input image is passed on to a multiple candidate word generation module which generates candidate words, each with a set of character regions and their corresponding unary potentials. With the help of an initial lexicon, pairwise priors are computed and diverse solutions are inferred from all the candidate words. These candidates are used to reduce the lexicon. This process is repeated with the reduced lexicon until the lexicon is refined to a small size. The final solution is the word in the full lexicon closest to the diverse solutions computed in the last iteration.

3.3.1 CRF framework

The CRF is defined over a set of random variables $x = \{x_i | i \in V\}$, where $V = \{1, 2, \dots, n\}$, denotes the set of n characters in a candidate word. Each random variable x_i denotes a potential character in the word, and can take a label from the label set L containing English characters and digits. The energy function, $E : L^n \rightarrow \mathbb{R}$, corresponding to a candidate word can be typically written as the sum of unary and pairwise potentials:

$$E(x) = \sum_{i \in V} E_i(x_i) + \sum_{(i,j) \in \mathcal{N}} E_{ij}(x_i, x_j), \quad (3.1)$$

where \mathcal{N} represents the neighbourhood system defined over the candidate word. The set of potential characters is obtained by a segmentation procedure, discussed in Section 3.3.2.

Unary Potentials. The unary potential of a node is determined by the SVM confidence score. The unary term $E_i(x_i = c_j)$ represents the cost of a node x_i taking a character label c_j , and is defined as:

$$E_i(x_i = c_j) = 1 - p(c_j | x_i), \quad (3.2)$$

where $p(c_j | x_i)$ denotes the likelihood of character class c_j for node x_i .

Pairwise Potentials. The pairwise cost of two neighbouring nodes x_i and x_j taking a pair of character labels c_i and c_j is defined as,

$$E_{ij}(x_i, x_j) = \lambda_l (1 - p(c_i, c_j)), \quad (3.3)$$

where $p(c_i, c_j)$ is the bi-gram probability of the character pair c_i and c_j occurring together in the lexicon. The parameter λ_l determines the penalty for a character pair occurring in the lexicon. Similar to [78],

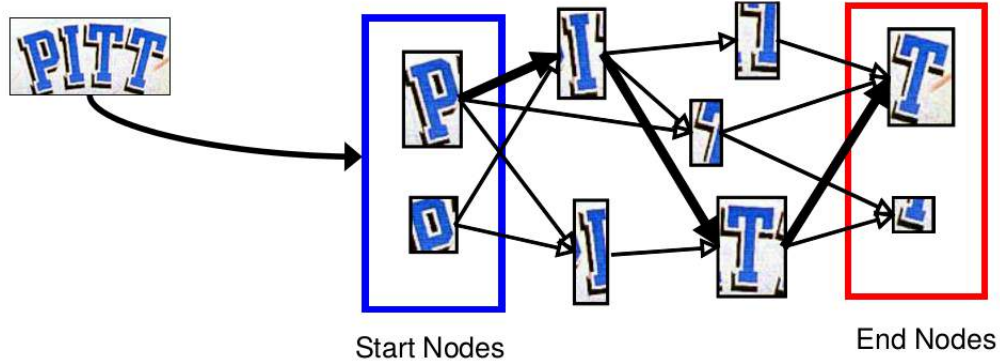


Figure 3.3 Visualization of candidate words generation on a sample image. A subset of possible character locations mark the start and end of the candidate words. Multiple combinations are generated based on adjacency factors of the character windows. The correct set of character windows forming a candidate word are shown with bold line connections.

we use node-specific prior, where the priors are computed independently for each edge from the bigrams in the lexicon that have the same relative position to that of the edge in the CRF. This enforces spatial constraints on prior computation, and are found to be more effective than the standard node prior [78].

3.3.2 Generating Candidate Words

Obtaining potential character locations with a high recall is desired for our approach. There are two popular methods for character extraction based on: (i) sliding window [77, 78], (ii) binarization [9, 120]. We follow the binarization based approach as it results in fewer potential character locations in the form of connected components (CCs) than those generated by the sliding window based method. This avoids redundant character windows, with similar size at a specific image location. Binarization based methods reduce candidate windows with threshold parameters and by leveraging fast pruning techniques on the CCs. To ensure that all the characters are present in the candidate windows as CCs, we combine results with different thresholds. This significantly improves the character recall at the cost of generating some false windows that can be overcome in the latter steps.

To remove obvious false windows we use heuristics based on information such as character sizes, aspect ratio and spatial consistency, followed by a character specific non-maximal suppression. This step removes false positive windows occurring in the background or unwanted foreground text elements like text bounding boxes. We also detect other anomalous windows, like holes in characters and invalid windows present within the characters, by finding configurations where a smaller window is contained completely within a larger window, and then remove the smaller one.

After pruning, we get a set of potential character windows which are used to generate candidate words. We first build a graph by joining the potential character windows which are spatially consistent and likely to be adjacent characters as shown in Figure 3.3.2. In other words, the windows are connected

with an edge if (i) overlapping windows have an overlap less than a threshold, and (ii) non-overlapping windows are less than a threshold away. We remove a few edges connecting windows whose width or height ratio is not in a desired range, to ensure that only character-to-character links are preserved. Then we estimate the most probable words for further analysis as described in the following.

Selection of Candidate Words. Our objective is to find a set of probable candidate words from the directed graph described above. We define a candidate word as a set of character windows representing the text present in the image. We first find the most probable start and end character windows by selecting windows close to image left and right boundary. Representing these start and end windows as candidate start and end nodes, we find possible connected paths (i.e., candidate words) between all pairs of start and end nodes using a depth first all paths algorithm [112]. We reject candidate words which do not cover sufficient area over the word image. The shortlisted candidate words are represented as a CRF, inferred and re-ranked according to their minimum energy value which is normalized by the number of nodes in the CRF. The least energy candidate words are retained for the subsequent stage as the correct candidate words assuming they have nodes with better unary potentials.

3.3.3 Diversity Preserving Inference

Once the optimal candidate words are selected, we infer the text each of them contains by minimizing the energy (3.1). However, the minimum energy solution of the word may be at times incorrect due to poor unary or pairwise potentials. Hence, diverse solutions are preferred over a single solution. Inspired by [8], we obtain M -best solutions instead of one MAP solution. This is done for all the selected candidate words from the previous stage individually. We approach the problem of diversity preserving inference with a greedy algorithm. First, we obtain the MAP solution with TRW-S [51] and then, the next solution is defined as the lowest energy state with minimum similarity from the previously obtained solutions.

Rewriting our optimization function (3.1) we obtain,

$$\min \sum_{i \in V} \alpha_i(s) \mu_i(s) + \sum_{i,j \in \mathcal{N}} \alpha_{ij}(s, t) \mu_{ij}(s, t), \quad (3.4)$$

where $\alpha_i(s)$ is the unary potential and $\alpha_{ij}(s, t)$ is the pairwise potential. The terms $\mu_i(s)$ and $\mu_{ij}(s, t)$ are their corresponding binary indicator variables with $s \in L$, the label set containing English characters and digits. Using a vector representation of $\boldsymbol{\mu} = \{\mu_i(s) | i \in V, s \in L\}$, the function (3.4) can be rewritten with standard constraints on unary and pairwise potentials as well as the diversity constraint (to get the second best solution) in the form of function $\Delta(\hat{\boldsymbol{\mu}}, \boldsymbol{\mu})$, where $\hat{\boldsymbol{\mu}}$ is the best solution found after inferring with the diversity constraint as follows,

$$\min \sum_{i \in V} \alpha_i(s) \mu_i(s) + \sum_{i,j \in \mathcal{N}} \alpha_{ij}(s,t) \mu_{ij}(s,t), \quad (3.5)$$

$$\text{s.t.} \quad \sum_{s \in L} \mu_i(s) = 1, \quad (3.6)$$

$$\sum_{s \in L} \mu_{ij}(s,t) = \mu_j(t), \quad \sum_{t \in L} \mu_{ij}(s,t) = \mu_i(s), \quad (3.7)$$

$$\Delta(\hat{\boldsymbol{\mu}}, \boldsymbol{\mu}) \geq k, \quad (3.8)$$

$$\mu_i(s), \mu_{ij}(s,t) \in \{0, 1\}. \quad (3.9)$$

Here, (3.6) and (3.7) denote the constraints on unary and pairwise potentials. The constraint (3.8) is the diversity measure that has to be greater than a scalar k . The Lagrangian relaxation of this optimization problem is formed by the dualizing the constraint (3.8), which yields,

$$\min \sum_{i \in V} \alpha_i(s) \mu_i(s) + \sum_{i,j \in \mathcal{N}} \alpha_{ij}(s,t) \mu_{ij}(s,t) - \lambda(\Delta(\hat{\boldsymbol{\mu}}, \boldsymbol{\mu}) - k). \quad (3.10)$$

Using a dot product dissimilarity (Hamming distance) as our Δ function we obtain,

$$\min \sum_{i \in V} \alpha_i(s) \mu_i(s) + \sum_{i,j \in \mathcal{N}} \alpha_{ij}(s,t) \mu_{ij}(s,t) - \lambda(-\hat{\boldsymbol{\mu}} \cdot \boldsymbol{\mu} - k), \quad (3.11)$$

which can be re-written as,

$$\min \sum_{i \in V} (\alpha_i(s) + \lambda \hat{\mu}_i(s)) \mu_i(s) + \sum_{i,j \in \mathcal{N}} \alpha_{ij}(s,t) \mu_{ij}(s,t) + \lambda \cdot k. \quad (3.12)$$

Hence, using the Hamming distance, only the unary potentials need to be modified by adding the original solution scaled by the diversity parameter λ . The TRW-S [51] algorithm can be utilized again to infer the second best solution.

3.3.4 Lexicon Reduction

Once the solutions are obtained from all the selected candidate words, they are used to reduce the large lexicon and compute pairwise potentials iteratively. We prefer to use the diverse solutions over the MAP solution as they maximize the chances of inferring the correct solution. Our first iteration involves shrinking the lexicon to a smaller size, i.e., 50. This is done by re-ranking the lexicon words using group edit distance (described below) to the solutions obtained, and retaining the top matches. This iteration reduces the lexicon size significantly and retains a small subset with a high recall of ground truth words. From the second iteration onwards, we use the new pairwise potentials and re-infer the diverse solutions. Thereafter, we remove the word in the lexicon with maximum group edit distance from the diverse solutions. This lexicon reduction procedure is summarized in Algorithm 2.

Algorithm 2: The lexicon reduction process alternates between removing words from the lexicon and re-computing the pairwise potentials.

Input: Candidate words, Initial lexicon L_i , Reduced lexicon size n

Output: Reduced lexicon L_r

Initialization: $L_r = L_i$

while $size(L_r) > n$ **do**

Step 1: Perform inference on all the candidate words to obtain M diverse solutions (Section 3.3.3)

Step 2: Remove the lexicon word w with the maximum group edit distance from M diverse solutions

$$L_r = L_r - \{w\}$$

Step 3: Compute new pairwise priors from the reduced lexicon

Lexicon →	STARS	THIS	TAP	...
Diverse solutions ↓				
TARS	1	2	2	...
TOLS	3	2	3	...
THIS	3	0	3	...
Group Edit Distance	1	0	2	...
Rank →	2	1	3	...

Table 3.1 Example of group edit distance computation between diverse solutions and lexicon words for the word image containing the word "THIS". We see that the selecting the lexicon word with less group edit distance from the diverse solutions works better as compared to the minimum energy solution "TARS".

Group Edit Distance. The standard way of re-ranking a lexicon using a single solution is by computing the edit distance between the solution and all the lexicon words. However in a multiple solution scenario, where diverse solutions from multiple words come into the picture, the correct inferred label is most likely to be present in the solution set. To be able to compute the edit distance between a solution set and lexicon, we find the minimum edit distance for each lexicon word from the solution set. This modification ensures that if the ground truth is very close to one of the diverse solutions, it will be ranked higher than others in the lexicon. A working example can be seen in Table 3.1 where the correct word "THIS" present in the lexicon, is eventually assigned its correct edit distance from the diverse solutions.

3.4 Recognition and Retrieval

The method described so far reduces the size of the lexicon by alternating between the two steps of estimating candidate words and refining the lexicon. We then use this lexicon for the recognition and retrieval tasks.

Recognition. In the recognition task, our goal is to associate a text label to a given word image. The process begins by forming multiple candidate words using the graph construction described in section 3.3.2. Candidate words are re-ranked and k optimal candidate words are retained. We reduce the lexicon (using the method in section 3.3.4) to a size of 10 words and obtain diverse solutions with the newly computed pairwise potentials from this reduced lexicon. We now select a word from the original lexicon with the minimum group edit distance from the diverse solutions as our result.

Retrieval. In a retrieval task, our objective is to retrieve word images for a given text query word from a dataset. The traditional approach would be to reduce the lexicon for each word to size one (hereafter referred to as singleton lexicon), and search for the query word in the singleton lexicon of all words in the dataset. However, since this approach is prone to recognition failure, we relax the constraint of reducing the lexicon to size one, and instead reduce the lexicon to a very small size, say five words. This allows us to overcome recognition errors and retrieve word images where the ground truth is present in the reduced lexicon but not in the singleton lexicon. This relaxation is allowed only for word images with lexicons having high similarity amongst its constituent words. We measure the similarity of words in the lexicon with a measure called Average Edit Distance (AED) which is defined as,

$$\text{AED} = \frac{1}{P} \sum_{w_i, w_j \in L_P} ED(w_i, w_j), \quad (3.13)$$

where L_P is the lexicon with P words and $ED(w_i, w_j)$ is the edit distance between words w_i and w_j . A low AED implies that the reduced lexicon has similar words and hence, one more lexicon reduction iteration may result in arbitrary loss of ground truth from the reduced lexicon. On the other hand, in cases with high AED score, the words in the reduced lexicon are different from each other.

As a preprocessing step to our retrieval task, we prepare the dataset by reducing the lexicons for each word image to either a singleton or a reduced lexicon. This is done by checking the AED score at a specific iteration of the lexicon reduction process. If the score is found to be less than θ (i.e. showing high similarity amongst the words in lexicon) we terminate the lexicon reduction process and associate the reduced lexicon of size n with the word image. We continue the process to get a singleton lexicon otherwise. In our retrieval task, for a given query word, we find all word images in the dataset which have the query word in their respective singleton or reduced lexicons. All selected word images are then ranked using a combined score from the following two criteria: (i) the lexicon size (one or n), and (ii) position of the query word in the lexicon ranked by group edit distance from the most recent diverse solutions in lexicon reduction process. A combined score is generated by summing up the lexicon size and the position of the query word in the lexicon. The word images with a lower combined score are ranked higher than others. We give more weightage to the first criterion, as word images with singleton lexicons are more likely to get inferred correctly and retain only the ground truth in their respective singleton lexicons. On the other hand, word images with reduced lexicons have additional words besides the ground truth which may lead to incorrect retrievals and hence must be ranked lower. The second

criterion helps us in ranking word images with the same score from first criterion as the word images with the query words at a higher position in the lexicon are more likely to be the groundtruth.

3.5 Experimental Analysis

3.5.1 Datasets

We used three public datasets, namely IIIT 5K-word dataset [77], ICDAR 2003 [1] and Street View Text (SVT) [2, 118] in our evaluations.

IIIT 5K-word. The IIIT 5K-word dataset contains 5000 cropped word images from scene texts and born-digital images, harvested from Google image search engine. This is the largest dataset for natural image word spotting and recognition currently available. The dataset is partitioned into train (2000 word images) and test (3000 word images) sets. It also comes with a large lexicon of 0.5 million words. Further, each word is associated with two smaller lexicons, one containing 50 words (known as small lexicon), another with 1000 words (known as medium lexicon).

ICDAR 2003. The test dataset contains 890 cropped word images. They were released as a part of the robust reading competitions. We use small lexicons provided by [118] of size 50 for each image for ICDAR 2003 dataset.

SVT. The SVT dataset contains images taken from Google Street View. Since we focus on the word recognition task, we used the SVT-word dataset, which contains 647 word images and a 50-word sized lexicon for each image.

3.5.2 Multiple Candidate Word Generation

We binarize the image using Otsu’s method [88] with ten thresholds equally spaced over the grayscale range. This provides a good set of potential character locations, which are used to construct the graph (Section 3.3.2). The overlap, aspect ratio and width/height range parameters associated with the graph construction are chosen by cross-validating on an independent validation set. We add an edge between two overlapping windows if their X-axis projection intersection is less than 25% of the left window width. If they are non-overlapping, they must be no farther than 50% of the left window width. We remove edges with window width ratio or height ratio more than a factor of 4. For non-maximal suppression, we use 80% overlap as our threshold. Once the graph is constructed, all candidate words are found (Section 3.3.2). We then re-rank them using their energy score (3.4) normalized by word length and select the top-10 candidate words for the lexicon reduction phase.

Word Image	Iteration 1	Iteration 2	Iteration 3	Iteration 4
	FGAIEESHER	FGAIERSHER	KINGFISHER	KINGFISHER
	NHAI	AHAI	AHAI	THAT
	MAITOTA	MAITOTA	MACTOTH	MAMMOTH
	THTL	THEL	THEL	THIS

Table 3.2 Effect of the lexicon reduction technique on the inferred label. Here we show four iterations for each example. We observe that with stronger pairwise potentials the method recovers from the errors in the MAP solution.

3.5.3 Diversity Preserving Inference

We train one-vs-all character classifiers with linear SVM for unary potentials, as described in [79], with dense HOG features [18] from character images. To obtain multiple CRF solutions we infer the top-5 diverse labels by modifying the unary potentials in each iteration (Section 3.3.3). The λ parameter in (3.12) is set by cross validation. We found $\lambda = 0.1$ to be the optimal parameter. We see that with a smaller λ , the unary potentials will be modified by a very small amount in the next iteration which will result in inference of the same solution with higher energy. Hence, to reach to a new solution, we would be requiring more number of iterations. On the other hand, with a larger λ , the unary potentials are modified to a good extent which lead to a solution significantly different from the previous solution. This leads to unfavourable diversity where words are significantly different from each other.

3.5.4 Recognition

In our recognition experiment, we stop the lexicon reduction process when the reduced lexicon reaches a size of 10, and then find the nearest word in the original lexicon with minimum group edit distance from the most recently inferred solution set. We evaluate the performance of the system by checking if the nearest word is ground truth or not.

For the large lexicon experiments, the group edit distance reranking becomes computationally expensive due to the lexicon size. To speed up the process, we represent each word by its character histogram and build a k -NN classifier. Now, for a given solution set and a lexicon, we first find the top-100 nearest neighbours in the lexicon for each word in the solution set. We then consider the union of all top-100 nearest lexicon words to be the new lexicon and perform the group edit distance based reranking on it. This speeds up the process by around 200 times and reduces the computation time to less than a second.

Discussion.

Table 3.2 shows that lexicon reduction (and re-computation of priors using diverse solutions) corrects solutions in the first four iterations. We observe that the inferred label change by one or more characters

Method	IIIT 5K-word			ICDAR 03	SVT
	Large	Medium	Small	Small	Small
non-CRF based					
Wang et al. [117]	-	-	-	76.0	57.0
Bissacco et al. [9]	-	-	-	82.8	90.3
Alsharif et al. [6]	-	-	-	93.1	74.3
Goel et al. [29]	-	-	-	89.6	77.2
Rodriguez et al. [94]	-	57.4	76.1	-	-
CRF based					
Shi et al. [101]	-	-	-	87.4	73.5
Novikova et al. [86]	-	-	-	82.8	72.9
Mishra et al. [78]	-	-	-	81.7	73.2
Mishra et al. [77]	28.0	55.5	68.2	80.2	73.5
Our Method	42.7	62.9	71.6	85.5	76.4

Table 3.3 Word recognition accuracy comparison between various CRF and non-CRF methods. A word is said to be correctly recognized if the nearest word in the lexicon to its solution by the method is the ground truth. We compute top-5 diverse solutions and select one solution from the full lexicon with minimum group edit distance as our proposed method. We see that in the large and medium lexicon setting of IIIT 5K-word dataset, our method outperforms the existing ones. We also obtain similar performance as compared to the other CRF methods on small lexicons.

as the priors get stronger over iterations by assigning a lower pairwise potential to the bigrams from the ground truth.

Table 3.3 compares the performance of the proposed method with the state of the art over the three datasets. We see that our method outperforms the state of the art in the large lexicon setting. We obtain 14% improvement over [77] because of stronger priors.¹ As a baseline, we searched for multiple candidate words using the CRF energy without using the diversity constraint. For example, on the IIIT 5K-word dataset (with medium lexicon), this resulted in an accuracy of 55.6 without diversity compared to 62.9 (with diversity, shown in Table 3.3), when considering the top-5 candidate words.

For the small lexicon setting, non-CRF methods, like beam search on a graph in [9] perform well on SVT dataset because of training the classifiers with millions of character images. This is around ten times larger than the amount of training data we use, and is unavailable to the public. The structured SVM formulation [94] shows a good performance on the small lexicon of IIIT 5K-word dataset but deteriorates as the lexicon size increases. This is due to the model being incapable of effectively minimizing the distance between the label and image features in the embedded space for larger lexicons.

¹It should also be noted that [77] follows an open vocabulary lexicon, i.e., it does not assume that the ground truth is present in the lexicon. We find that around 75% of the ground truth words from IIIT 5K-word dataset are present in the large lexicon by default. The rest of the ground truth words are language-specific and proper nouns like city and shop names.

K	ICDAR 03		ICDAR 11		ICDAR 13		IIIT 5K-word		SVT	
	M	L	M	L	M	L	M	L	M	L
	<i>Non Diverse</i>									
1	78.9	61.5	69.9	51.1	70.5	51.0	58.0	40.9	66.4	48.3
3	79.1	63.9	69.8	51.4	70.5	51.2	58.3	40.9	66.6	48.9
5	79.2	63.6	69.9	51.2	70.5	51.2	57.8	40.0	66.7	48.8
	<i>Diverse</i>									
1	77.0	62.7	68.2	52.3	69.4	52.6	57.7	38.9	67.2	51.4
3	78.3	66.9	70.3	57.2	70.6	57.1	62.2	43.9	67.2	51.4
5	80.0	66.5	70.0	58.1	72.1	59.0	62.9	45.3	67.3	51.4

Table 3.4 Word recognition accuracy comparison for the proposed method while varying the number of solutions considered represented by the parameter **K**. We see that diverse solutions do improve performance especially on large lexicons and in general, perform better than non-diverse solutions.

3.5.5 Retrieval

In this experiment we retrieve a word image for a given query word from the dataset. The dataset comprises of a singleton or a reduced lexicon for each image which is used for the task as described in section 3.4. As our proposed method, we preprocess the dataset by reducing the lexicon to singleton if the AED value θ at the 5th (last) iteration is less than 3.5. We call this process the *partial reduction* method as it reduces the lexicon to size one only for some word images, and for the rest, the lexicon contains 5 words. As a baseline method, we also do a *full reduction*, reducing lexicons for all the word images to one corresponding word. Both, the proposed and the baseline methods, are performed with and without the diversity constraint, thus creating four different variations. The parameter θ that gives the best precision for the proposed method is selected after cross validation over an independent query set. For quantitative evaluation, we compute the precision of the first retrieved word image as the datasets do not have a significant number of repeating ground truth labels (i.e., word images with the same text).

We show quantitative results in Table 3.5, where we clearly see that partial reduction of lexicons with diversity outperforms full reduction without diversity on the IIIT 5K-word dataset. The diverse solutions improve the performance as they retain the ground truth in reduced lexicon after the lexicon reduction process in many cases. We also notice that on IIIT 5K-word dataset, the performance gap increases as the lexicon size increases, suggesting potential applicability to larger lexicon based query systems. Correct retrievals (in Fig. 3.4) show that a higher AED threshold based lexicon association has the ground truth in the reduced lexicon associated with it, as compared to its singleton lexicon. The method is less successful in cases (Figure 3.5) where the ground truth is lost in the early stages of lexicon reduction leading to a reduced lexicon without the ground truth in it. This happens due to failure of the binarization method used to segment out the characters, which leads to abrupt short/long candidate word formation.







Query	Retrieved Image	Reduced Lexicon with diversity and partial reduction	Lexicon without diversity and full reduction
BRADY		MY, BRADY, ANY, A	MY
SPACE		HOT, SPACE, LACEY, SALE	HOT
HAHN		BUENA, HANDA, HAHN, PIPE	BUENA
DAILY		PEARL, MOUNTS, DAILY, NIKE	PEARL
TIMES		TIME, TIMES, WINE, MED	TIME
THREE		THE, THREE, THERE, USED	THE

Figure 3.4 Cases where retrieval results are correct. The reduced lexicon from partial reduction method retains the ground truth word. The words in the reduced lexicon are similar to each other, and any further reduction could have resulted in loss of ground truth.







Query	Retrieved Image	Reduced Lexicon with Proposed Method
CLEAR		CLEAR
HOME		HOME, 900AM, 9080, 90
BAR		BAR
FOR		AND, ARTS, FOR, INN
311		311
JOIN		ONE, JOIN, OUT, OUR

Figure 3.5 Failure cases for retrieval experiment. Some word images have reduced lexicons with no ground truth. Other cases have the ground truth word, but are retrieved for the wrong query word (rows 2,6).

Method	IIIT 5K-word			ICDAR 03
	Large	Medium	Small	Small
Without diversity				
Full Reduction	27.5	51.9	65.0	81.7
Partial Reduction	35.1	35.6	60.7	76.9
With diversity				
Full Reduction	23.1	52.0	65.0	78.9
Partial Reduction	42.1	59.0	66.5	79.5

Table 3.5 Top-1 precision for retrieval experiment on various datasets. We compare the results between two reduction methods, each with and without diverse solutions. The partial reduction method leaves some lexicons with around 5 words, while the full reduction method reduces all lexicons to size one. We see that our proposed method of partial reduction with diverse solutions works the best for the IIIT 5K-word dataset.

3.6 Summary

In this chapter, we proposed a novel framework for recognition and retrieval tasks in the large lexicon setting. We identify potential character locations and find words contained in the image. We reduce the large lexicon to a small image-specific lexicon. The lexicon reduction process alternates between recomputing priors and refining the lexicon. We evaluated our results on public datasets and show superior performance on large and medium lexicons for recognition and retrieval tasks.

Chapter 4

End to End Frameworks

4.1 Introduction

Scene understanding can greatly benefit from identifying the text within the images. One way to analyse the text in scene images is through end to end frameworks which comprise of a text detection module to locate the text and a recognition module to recognize the text. The text content and location can provide useful information in many cases including indexing images for retrieval to robots understanding the scene.

End to end systems are challenging to build as they have to tackle both detection and recognition errors arising from the high amounts of variation in scene text. Several variants of such end to end frameworks exist relying on different kinds of text detection and recognition methods. Most common approach is to detect the text location in the form of bounding boxes and feed it to the recognition system. The recognition systems in these cases can be holistic or character wise in nature. A variant of such system is to use the text segmentation as an input to the recognition module which can work on a binarized image to identify the words present. Another approach to develop an end to end framework is to use a text region proposal method and a recognition module to recognize the proposals and prune the results. Such frameworks can be slightly advantageous if we have a strong recognition module which can be use to correct the detection results, hence providing a feedback to the detection system.

In this work, we utilize both the text detection systems as discussed in chapter 2 and the CRF based recognition module from chapter 3 to build end to end recognition and retrieval frameworks. The coupling strategies are discussed in section 4.3 and experiment results on popular datasets are presented in section 4.4.

4.2 Related Work

Work on developing an end to end recognition framework for scene images began in late 90s [14, 26, 41, 126] but relied on OCRs for recognition. Such systems utilized existing OCR techniques on scene text which possessed higher amounts of variation. The first end to end recognition pipeline without

OCRs was proposed by Neumann et al. [81] which classified MSERs using a SVM on hand-crafted features to find character candidates. Thereafter, text lines were found and geometrically normalized to remove perspective distortion and eventually recognized with the help of a character model. A sliding window based method was proposed by Wang et al. [116] where a random ferns character classifier was slid across the image to get candidate regions. Pictorial structures were used to find words from a lexicon in the end. Coates et al. [17] proposed a unsupervised feature learning method for text and character classification by mining meaningful patches from the dataset and synthesizing descriptors for the classifiers. The MSER pruning method based detection method in [82] was improved with fast incremental descriptors for character recognition in [83] providing a real time end to end system. A single random forest classifier for detection and recognition was presented in [129] which relied on dictionary search to correct recognition results.

PhotoOCR [9] proposed a deep learning based character classifier trained on 11 million characters which achieved an classification accuracy of 92% on the ICDAR 2003 dataset. The system used multiple complementary text detection schemes to generate text lines which were over-segmented and then merged to form meaningful words using a beam search based technique. Another CNN based classification and detection was proposed by Jaderberg et al. [40] where multiple detection results were generated and corrected by the recognition module, thus achieving end to end recognition results.

Very few end to end retrieval pipelines have also been proposed till now. Scene image retrieval using text queries was first proposed by Mishra et al. [79] where the authors use a sliding window character classifier and compute scores for each query based on spatial information and ordering of the characters. Jaderberg et al. [39] extend the work using text region proposals and a CNN based recognition engine and obtain significant improvement in performance.

4.3 Proposed Method

In this section, we propose end to end frameworks for scene text using the detection methods presented in chapter 2 and the recognition system from chapter 3. Our objective is to localize and recognize the text in an image in two settings, (i) without a lexicon by combining the text segmentation method with Tesseract and, (ii) with a lexicon by combining region proposals with the CRF based recognition module.

4.3.1 Text Segmentation + Tesseract

In this pipeline, we use the single linkage hierarchical clustering (SLC) based method to segment text from image. A binarized image is formed using the MSERs present in clusters classified as text by the adaboost text/non-text classifier. The binarized image is considered as an page level input to a popular printed text OCR called Tesseract [109] which returns word level bounding boxes and the recognized text. The MSERs present on the binarized image give correct word level segmentation

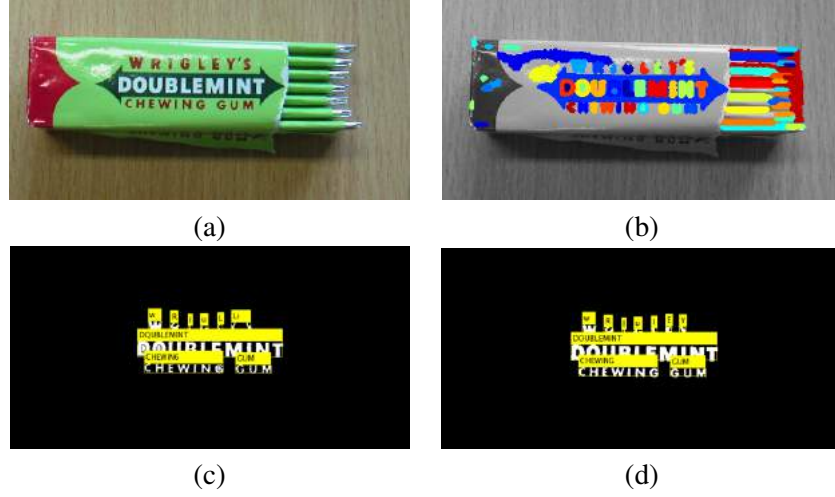


Figure 4.1 End to end pipeline for text segmentation + Tesseract. We begin by finding MSERs on the grayscale (b) of the original image (a). The text classified MSER clusters are used to form a binarized image which is fed to the OCR to retrieve initial end to end result (c). The OCR is able to perform a page layout and reject components in the binarized image which are not text. The detected text has correct location but incorrect recognition result i.e. "DQUBLEMINT" and "CHEWIN6". The word region is locally thresholded by otsu and the whole image is given back to the OCR, thus correcting the recognition result to "DOUBLEMINT" and "CHEWING".

but may effect recognition due its slight difference from the actual characters present in the image which effects the OCR's character prediction. Hence, as a last step, we improve the recognition output by locally binarizing the word segmentations with otsu [88] which is processed again by the OCR to generate the final result. We also implement a garbage filter to remove words with only spaces or oddly repeating characters like i. An overview of the pipeline can be seen in Figure 4.1.

4.3.2 Region Proposals + CRF based Recognition

In this pipeline, the CNN based region proposal method is used to generate several word candidates per scene image. Treating each word candidate as a cropped word region, we recognize them with the help of a lexicon which is iteratively reduced to one. Thereafter, we compute a single score from the detection and recognition output which is used to reject false positive detections by thresholding. The single score consists of a linear combination of the minimum energy value after the last iteration of the lexicon reduction process, the edit distance with respect to the closest word in the lexicon and the mean CNN score for the region computed using the score map.

4.3.3 Scene Image Retrieval

Given a text query, we use the CNN+CRF pipeline to retrieve images. During indexing, we recognize each proposal using the query set as our lexicon and associate a linear combination of the minimum edit

distance, minimum energy and mean CNN score to them. At retrieval stage, for a given query we find the images with proposals having the recognized label as the query and re-rank them using the scores. We retain all proposals for retrieval, leveraging on the high recall for optimal performance.

4.4 Experimental Analysis

4.4.1 Datasets

We use the scene images from the various ICDAR datasets as stated in section 2.4. For each dataset, we consider three lexicon sizes i.e. *small* comprising of 50 words, *full* comprising of all ground truth in the dataset. For OCR based recognition, no lexicon is used to correct the results.

4.4.2 Evaluation

We consider a correct word detection and recognition only if the corresponding detected bounding box has an overlap of more than a threshold with respect to the ground truth and has the text correctly recognized. In case of lexicons, we consider a correct recognition if the method is able to select the ground truth from the lexicon. For OCR based recognition, we use case sensitive matching of ground truth and recognition results. We calculate recall by finding the percentage of ground truth words correctly identified in the scene image and precision by the percentage of detections which are ground truth. For retrieval results, we use the standard average precision and mean average precision for a quantitative analysis.

4.4.3 Text Segmentation + Tesseract

The segmentation via clustering + Tesseract approach (referred as SLC + Tesseract) is evaluated on the ICDAR 2011 scene images dataset in Table 4.1. The end to end performance of a word is considered as detection with overlap more than 80% with case sensitive word label matching. We outperform a similar pipeline comprising of CSERs and Tesseract [33] where input to the OCR is word level regions. We rely on page level input to Tesseract, hence making effective use of its line and word segmentation methods for horizontal text while rejecting any misclassified non-text component. Few qualitative results where the method performs well are shown in Figure 4.3.

4.4.4 Region Proposals + CRF based Recognition

The CNN+CRF method is evaluated on the ICDAR 2003 dataset with small and full lexicons in Figure 4.2(b). The small lexicon contains 50 words and the full lexicon contains 860 words. The end to end performance is measured by considering detection with overlap more than 50% and correct word being picked from the lexicon for recognition. We linearly combine the mean CNN score, CRF energy

Method	Recall	Precision	F measure
Neumann et al. [83]	37.2	37.1	36.5
Wang et al. [116] ¹	30.0	54.0	38.0
Neumann et al. [84]	39.4	37.8	38.6
<i>Tesseract based</i>			
Gomez et al. [33]	32.4	52.9	40.2
SLC + Tesseract	32.0	59.0	41.5

Table 4.1 Lexicon free end to end framework performance of various methods on ICDAR 2011 dataset. Our SLC + Tesseract method performs better than the sliding window based method [116] as well as the other MSER based methods with heuristic based grouping of characters [83, 84].

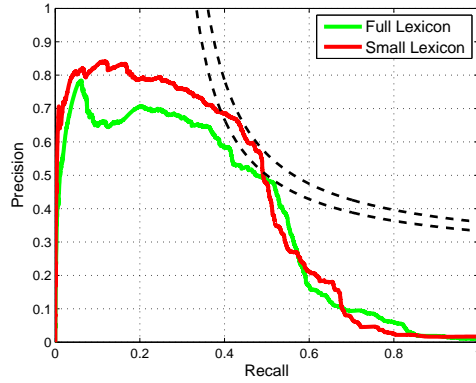
and minimum edit distance using the weights $(-0.1, 0.1, 0.9)$ for small lexicon and $(-0.1, 0.2, 0.7)$ for full lexicon which is found experimentally on a validation set. Figure 4.2(a) plots the precision recall curve for both the lexicons.

We obtain a recall of 47 and 52 with a precision of 60 and 48 for small and full lexicon respectively using the CNN+CRF method. We observe this method is outperformed by other methods on both the lexicons especially by the CNN detection and recognition based end to end systems by Jaderberg et al. [39, 40] which reach an f-measure of 90%. We find that even though the CNN+CRF method has pairwise terms computed from the lexicon for the CRF, it does not rely on any spatial information about the characters individually on the word image while generating the score. The mean CNN detection score only informs us about the probability of text in the proposed region and the CRF minimum energy score only considers the unary and pairwise terms computed after the characters are localized in the proposed word region. This severely effects the performance showing that the character level spatial information in the form of similar character sizes and foreground, uniform spacing between the characters, etc. is an important cue for rejection of false positive proposals which do not contain any words in them.

4.4.5 Scene Image Retrieval

We evaluate the CNN+CRF method for retrieval on the ICDAR 2011 dataset. We use the weights $(-0.1, 0.5, 1.5)$ to combine the mean CNN score, minimum energy and minimum edit distance to get a single score for each proposal. We obtain an average precision of 42% for the top-1 result with a mean average precision (mAP) of 28%. The approach is outperformed by Mishra et al. [79] which obtains a mAP of 65.3% and the CNN based method of Jaderberg et al. [39] with a mAP of 90.3%. As described in the previous section, the CNN+CRF method is limited by the absence of a spatial information of the characters in the scoring function as well as its dependency on the edit distance score which is less likely to perform well in case of several proposals. Successful retrievals for five text queries on the ICDAR 2011 dataset is shown in Figure 4.4.

¹Evaluated on the slightly different ICDAR 2003 dataset



(a)

Method	Small	Full
Wang et al. [116]	68	51
Alsharif et al. [6]	77	70
Jaderberg et al. [40]	80	75
Jaderberg et al. [39]	90	86
CNN+CRF	53	50

(b)

Figure 4.2 Lexicon based end to end framework performance of various methods on ICDAR 2003 dataset. (a) shows the precision and recall curve using the combined score of the CNN+CRF method for each lexicon size with their respective weights. Curves with constant f-measure are shown in dashed lines. Table (b) compares the f-measure with the state of the art methods.

4.5 Summary

In this chapter, we discuss two contrasting end to end recognition pipelines to recognize text on scene images and a retrieval pipeline for scene images. The first pipeline is based on detecting text by segmenting the text pixels and relying on Tesseract OCR for word level segmentation and a lexicon free recognition. The second pipeline consists of generating several word region proposals which are then evaluated by a CRF based word recognition module with a given lexicon. The second pipeline bypasses explicit detection of text as well as relies on a lexicon for correction of results. This pipeline is also used for retrieval of scene images, without pruning the proposals. We perform experiments on standard datasets and compare with successful works of this decade for both of these approaches.

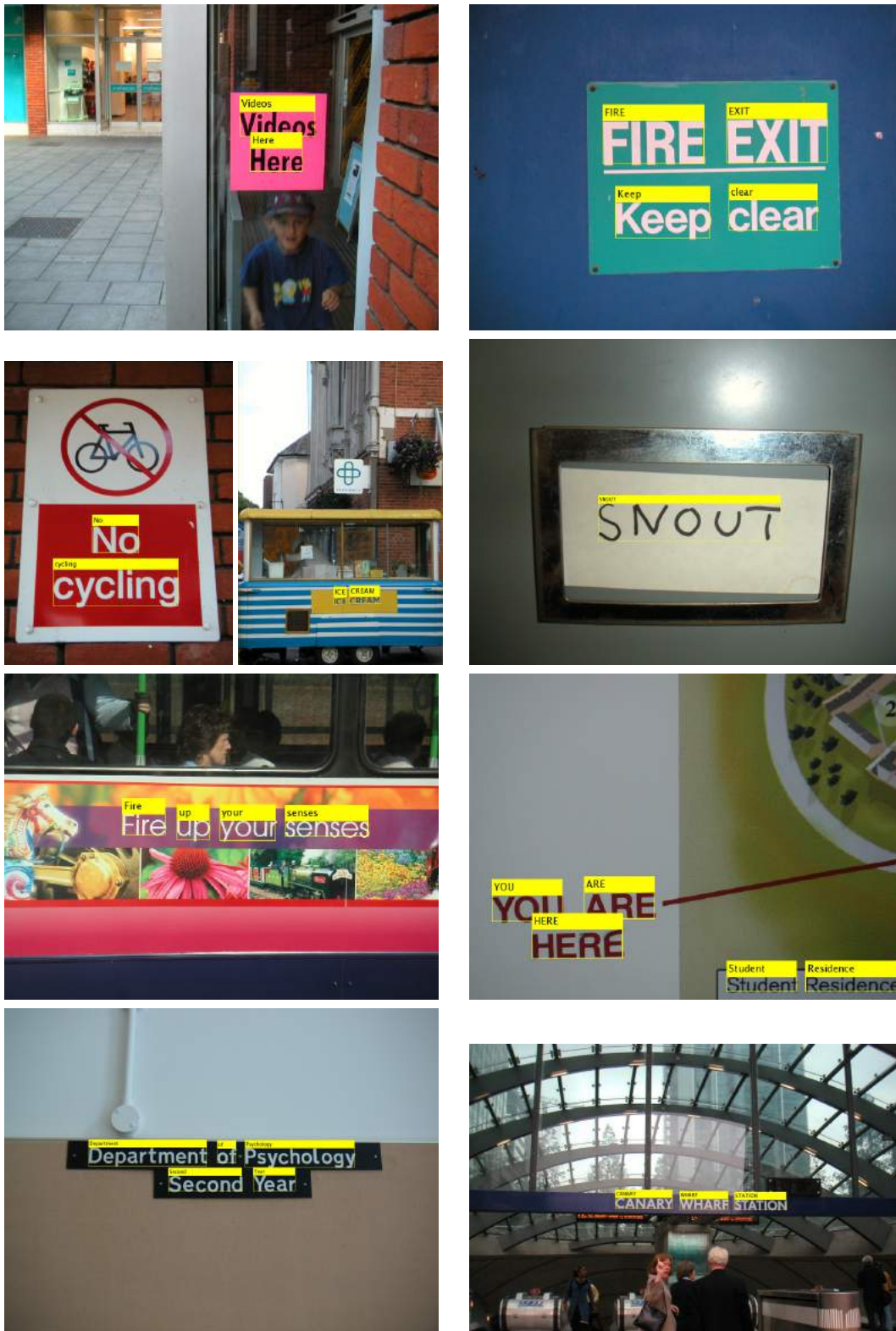


Figure 4.3 Successful qualitative results for the SLC + Tesseract method. Our method is able to correctly segment out the text in presence of non-text elements and recognize them. A standard OCR is able to recognize simple text with single foreground color and appropriate inter word spacing.



Figure 4.4 Qualitative scene image retrieval results for few text queries on the ICDAR 2011 dataset. Each column shows the query text and the top 2 image retrievals using the CNN+CRF end to end pipeline.

Chapter 5

Conclusions

In this thesis, we proposed solutions to standard challenges in scene text analysis. We discussed and implemented two kinds of text detection methods in Chapter 2 and improved the performance of lexicon based cropped word recognition, especially for large lexicons comprising of 0.5 million words in Chapter 3. Finally, we combine the detection and recognition modules, resulting in two contrasting end to end pipelines which are discussed in Chapter 4.

For the text detection challenge, we pursue a detection via segmentation approach using hierarchical clustering. We find MSERs on the scene image which are clustered using the single linkage clustering method, thus generating several overlapping text candidates. The candidates are classified using a text/non-text adaboost classifier and the positively classified ones are used to create a binarized image consisting of text pixels as foreground. We also implement a text region proposal method which generates several possible word level regions on the scene image with high recall. The method relies on a patch level text/non-text CNN classifier which generates a score map with per pixel probability of text occurrence. The score map is thresholded and components are combined using RLSA to generate the proposals.

For the text recognition task, we improved an existing CRF based framework for word recognition using a lexicon. We generated several word candidates through multiple binarization of the cropped word image and represented each of them in a CRF framework with characters as nodes. Diverse solutions were inferred from the the word candidates using a pairwise prior computed from the lexicon associated which in turn was used to reduce the lexicon. The process of inference on word candidates and lexicon reduction was done iteratively, thus reducing the lexicon to a single word. We also used the iterative lexicon reduction method to enable retrieval on cropped word images. Given, a set of cropped word images with a lexicon, we partially reduced the lexicon, thus associating a small set of words to each image with high probability of ground truth amongst them. At retrieval stage, we searched for our text query among the associated lexicons and ranked the retrieved word images based on edit distance and energy scores. We found our method to perform well in case of large lexicons and compared with other state of the art methods.

For the end to end pipelines, we built two pipelines combining different detection and recognition modules. The first pipeline consisted of the detection via segmentation approach using single linkage clustering on MSERs whose output was sent to the Tesseract OCR. The OCR word level recognition was further improved by binarizing the detected word region and running the OCR again at page level. The second pipeline consisted of the generating several region proposals using the CNN classifier and recognizing them using the CRF based framework. The recognition scores, with the help of a lexicon, were used to reject incorrect detections thus refining the detection as well as providing us with the recognition result.

5.1 Future Work

In the last two decades, scene text methods have evolved tremendously, utilizing script specific information and several vision techniques with increasingly powerful classifiers. The state of the art approaches are capable of detecting and recognizing English text with sufficient accuracy.

In case of detection, one direction to proceed with is to shift focus to script independent features which are common to majority of the scripts including handwriting. They can be then used during the grouping stage to cluster similar components. The assumption of detecting characters as separate components has to be relaxed, allowing detection of text from indic and arabic scripts to handwritings. Non-text content is also one area which requires attention by the community. Strategic layout templates can be devised to keep the non-text content out of the processing loop. These templates, for example can state the type of text they are looking for in a specific background, hence allowing us to define a region of interest in the image with the probable text.

Recognition methods require to increase the lexicon to the size of a dictionary. The lexicon needs to be a direct part of the solution ensuring a real-time solution by avoiding the costly DTW operation to find the best one. Holistic methods via deep learning or manifold embedding can be explored to avoid computationally expensive character wise searches. Multi-script recognition via unicodes is also an attractive field as most of the recognition datasets contain only English. For example, indic scripts like Hindi pose a different challenge when it comes to detecting the various consonants and vowels as well as the placement of the joiners.

Owing to the improved accuracies of the lexicon based recognition approach, scene text image retrieval solutions have been considering the query set as a lexicon and using them at the indexing stage. Solutions can be developed for an ideal retrieval scenario where the images are recognized and indexed without the knowledge of queries in the dataset. Efforts can be made to improve the estimation of posterior probability of the document being relevant given the query during retrieval phase.

Related Publications

Publications from this thesis,

- Udit Roy, Anand Mishra, Karteek Alahari, and C. V. Jawahar “*Scene Text Recognition and Retrieval for Large Lexicons*” In Asian Conference on Computer Vision (ACCV), 2014

Other publications during MS which are not a part of this thesis,

- Udit Roy, Naveen Sankaran, K. Pramod Sankar, and C. V. Jawahar. “*Character N-Gram Spotting on Handwritten Documents using Weakly-Supervised Segmentation.*” In International Conference on Document Analysis and Recognition (ICDAR), 2013
- Udit Roy, Tejaswinee Kelkar, and Bipin Indurkha “*TrAP: An Interactive System to Generate Valid Raga Phrases from Sound-Tracings*” In New Interfaces for Musical Expression (NIME), 2014

Bibliography

- [1] ICDAR 2003 datasets, <http://algoval.essex.ac.uk/icdar>.
- [2] Street View Text dataset, <http://vision.ucsd.edu/~kai/svt>.
- [3] M. Agrawal and D. Doermann. Clutter noise removal in binary document images. In *ICDAR*, 2009.
- [4] B. Al-Badr and S. A. Mahmoud. Survey and bibliography of arabic optical text recognition. *Signal Processing*, 1995.
- [5] B. Alexe, T. Deselaers, and V. Ferrari. Measuring the objectness of image windows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012.
- [6] O. Alsharif and J. Pineau. End-to-end text recognition with hybrid hmm maxout models. *arXiv*, 2013.
- [7] P. Arbeláez, J. Pont-Tuset, J. Barron, F. Marques, and J. Malik. Multiscale combinatorial grouping. In *CVPR*, 2014.
- [8] D. Batra, P. Yadollahpour, A. Guzman-Rivera, and G. Shakhnarovich. Diverse m-best solutions in markov random fields. In *ECCV*, 2012.
- [9] A. Bissacco, M. Cummins, Y. Netzer, and H. Neven. Photoocr: Reading text in uncontrolled conditions. In *ICCV*, 2013.
- [10] H. Bunke. Recognition of cursive roman handwriting: past, present and future. In *ICDAR*, 2003.
- [11] M. Cai, J. Song, and M. R. Lyu. A new approach for video text detection. In *ICIP*, 2002.
- [12] D. Chen, J.-M. Odobez, and H. Bourlard. Text detection and recognition in images and video frames. *Pattern Recognition*, 2004.
- [13] X. Chen, J. Yang, J. Zhang, and A. Waibel. Automatic detection and recognition of signs from natural scenes. *IEEE Transactions on Image Processing*, 2004.
- [14] X. Chen and A. L. Yuille. Detecting and reading text in natural scenes. In *CVPR*, 2004.
- [15] K. Chinnasarn, Y. Rangsanteri, and P. Thitimajshima. Removing salt-and-pepper noise in text/graphics images. In *IEEE Asia-Pacific Conference on Circuits and Systems*, 1998.
- [16] A. Clavelli, D. Karatzas, and J. Lladós. A framework for the assessment of text extraction algorithms on complex colour images. In *DAS*, 2010.
- [17] A. Coates, B. Carpenter, C. Case, S. Satheesh, B. Suresh, T. Wang, and A. Y. Ng. Text Detection and Character Recognition in Scene Images with Unsupervised Feature Learning. In *ICDAR*, 2011.

- [18] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In *CVPR*, 2005.
- [19] V. C. Dinh, S. S. Chun, S. Cha, H. Ryu, and S. Sull. An efficient method for text detection in video based on stroke width similarity. In *ACCV*, 2007.
- [20] D. Doermann. The indexing and retrieval of document images: A survey. *Computer Vision and Image Understanding*, 1998.
- [21] D. Doermann, J. Liang, and H. Li. Progress in camera-based document image analysis. In *ICDAR*. IEEE, 2003.
- [22] P. Dollár, R. Appel, S. Belongie, and P. Perona. Fast feature pyramids for object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014.
- [23] I. Endres and D. Hoiem. Category-independent object proposals with diverse ranking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014.
- [24] B. Epshtein, E. Ofek, and Y. Wexler. Detecting text in natural scenes with stroke width transform. In *CVPR*, 2010.
- [25] N. Ezaki, M. Bulacu, and L. Schomaker. Text detection from natural scene images: towards a system for visually impaired persons. In *ICPR*, 2004.
- [26] J. Gao and J. Yang. An adaptive algorithm for text detection from natural scenes. In *CVPR*, 2001.
- [27] B. Gatos, I. Pratikakis, and S. J. Perantonis. Adaptive degraded document image binarization. *Pattern recognition*, 2006.
- [28] J. Gllavata, R. Ewerth, and B. Freisleben. Text detection in images based on unsupervised classification of high-frequency wavelet coefficients. In *ICPR*, 2004.
- [29] V. Goel, A. Mishra, K. Alahari, and C. V. Jawahar. Whole is Greater than Sum of Parts: Recognizing Scene Text Words. In *ICDAR*, 2013.
- [30] L. Gomez and D. Karatzas. Multi-script text extraction from natural scenes. In *ICDAR*, 2013.
- [31] L. Gomez and D. Karatzas. A Fast Hierarchical Method for Multi-script and Arbitrary Oriented Scene Text Extraction. *arXiv*, 2014.
- [32] L. Gómez and D. Karatzas. Mser-based real-time text detection and tracking. In *ICPR*, 2014.
- [33] L. Gómez and D. Karatzas. Scene text recognition: No country for old men? In *ACCV Workshop*, 2014.
- [34] V. Govindan and A. Shivaprasad. Character recognitiona review. *Pattern Recognition*, 1990.
- [35] L. Hollink, G. Schreiber, J. Wielemaker, B. Wielinga, et al. Semantic annotation of image collections. In *Knowledge capture*, 2003.
- [36] J. Hosang, R. Benenson, and B. Schiele. How good are detection proposals, really? *arXiv*, 2014.
- [37] W. Hu, N. Xie, L. Li, X. Zeng, and S. Maybank. A survey on visual content-based video indexing and retrieval. *IEEE Transactions on Systems, Man, and Cybernetics*, 2011.
- [38] W. Huang, Y. Qiao, and X. Tang. Robust scene text detection with convolution neural network induced mser trees. In *ECCV*. 2014.

- [39] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman. Reading text in the wild with convolutional neural networks. *International Journal of Computer Vision*, 2014.
- [40] M. Jaderberg, A. Vedaldi, and A. Zisserman. Deep features for text spotting. In *ECCV*, 2014.
- [41] A. K. Jain and B. Yu. Automatic text location in images and video frames. *Pattern Recognition*, 1998.
- [42] C. Jung, Q. Liu, and J. Kim. A stroke filter and its application to text localization. *Pattern Recognition Letters*, 2009.
- [43] K. Jung, K. I. Kim, and A. K. Jain. Text information extraction in images and video: a survey. *Pattern Recognition*, 2004.
- [44] L. Kang, Y. Li, and D. Doermann. Orientation robust text line detection in natural images. In *CVPR*, 2014.
- [45] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. Gomez i Bigorda, S. Robles Mestre, J. Mas, D. Fernandez Mota, J. Almazan, and L.-P. de las Heras. Icdar 2013 robust reading competition. In *ICDAR*, 2013.
- [46] D. Katz. *Gestalt psychology, its nature and significance*. 1979.
- [47] E. Kim, S. Lee, and J. Kim. Scene text extraction using focus of mobile camera. In *ICDAR*, 2009.
- [48] K. I. Kim, K. Jung, and J. H. Kim. Texture-based approach for text detection in images using support vector machines and continuously adaptive mean shift algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2003.
- [49] W. Kim and C. Kim. A new approach for overlay text detection and extraction from complex video scene. *IEEE Transactions on Image Processing*, 2009.
- [50] K. Kise, A. Sato, and M. Iwata. Segmentation of page images using the area voronoi diagram. *Computer Vision and Image Understanding*, 1998.
- [51] V. Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2006.
- [52] H. I. Koo and D. H. Kim. Scene text detection via connected component clustering and nontext filtering. *IEEE Transactions on Image Processing*, 2013.
- [53] D. Kumar, M. N. A. Prasad, and A. G. Ramakrishnan. Multi-script robust reading competition in icdar 2013. In *International Workshop on Multilingual OCR*, 2013.
- [54] H. Li, D. Doermann, and O. Kia. Automatic text detection and tracking in digital video. *IEEE Transactions on Image Processing*, 2000.
- [55] Y. Li, W. Jia, C. Shen, and A. van den Hengel. Characterness: an indicator of text in the wild. *IEEE Transactions on Image Processing*, 2014.
- [56] J. Liang, D. Doermann, and H. Li. Camera-based analysis of text and documents: a survey. *International Journal of Document Analysis and Recognition*, 2005.
- [57] R. Lienhart and W. Effelsberg. Automatic text segmentation and text recognition for video indexing. *Multimedia Systems*, 2000.

- [58] R. Lienhart and A. Wernicke. Localizing and segmenting text in images and videos. *IEEE Transactions on Circuits and Systems for Video Technology*, 2002.
- [59] C. Liu, C. Wang, and R. Dai. Text detection in images based on unsupervised classification of edge-based features. In *ICDAR*, 2005.
- [60] Q. Liu, C. Jung, S. Kim, Y.-T. Moon, and J.-H. Kim. Stroke filter for text localization in video images. In *IEEE International Conference on Image Processing*, 2006.
- [61] Y. Liu, D. Zhang, G. Lu, and W.-Y. Ma. A survey of content-based image retrieval with high-level semantics. *Pattern Recognition*, 2007.
- [62] L. M. Lorigo and V. Govindaraju. Offline arabic handwriting recognition: a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2006.
- [63] S. M. Lucas. Icdar 2005 text locating competition results. In *ICDAR*, 2005.
- [64] S. M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, and R. Young. Icdar 2003 robust reading competitions. 2003.
- [65] J. Luo, D. Joshi, J. Yu, and A. Gallagher. Geotagging in multimedia and computer vision: a survey. *Multimedia Tools and Applications*, 2011.
- [66] M. R. Lyu, J. Song, and M. Cai. A comprehensive method for multilingual video text detection, localization, and extraction. *IEEE Transactions on Circuits and Systems for Video Technology*, 2005.
- [67] S. Manen, M. Guillaumin, and L. V. Gool. Prime object proposals with randomized prim’s algorithm. In *ICCV*, 2013.
- [68] S. Mao, A. Rosenfeld, and T. Kanungo. Document structure analysis algorithms: a literature survey. In *Electronic Imaging*, 2003.
- [69] K. Marukawa, T. Hu, H. Fujisawa, and Y. Shima. Document retrieval tolerating character recognition error evaluation and application. *Pattern Recognition*, 1997.
- [70] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 2004.
- [71] T. Mei, Y. Rui, S. Li, and Q. Tian. Multimedia search reranking: A literature survey. *ACM Computing Surveys*, 2014.
- [72] C. Merino-Gracia, K. Lenc, and M. Mirmehdi. A head-mounted device for recognizing text in natural scenes. In *CBDAR*. 2012.
- [73] S. Milyaev, O. Barinova, T. Novikova, P. Kohli, and V. Lempitsky. Image binarization for end-to-end text understanding in natural images. In *ICDAR*, 2013.
- [74] R. Minetto, N. Thome, M. Cord, N. J. Leite, and J. Stolfi. Snoopertrack: Text detection and tracking for outdoor videos. In *ICIP*, 2011.
- [75] R. Minetto, N. Thome, M. Cord, N. J. Leite, and J. Stolfi. T-hog: An effective gradient-based descriptor for single line text regions. *Pattern Recognition*, 2013.
- [76] A. Mishra, K. Alahari, and C. Jawahar. Image retrieval using textual cues. In *ICCV*, 2013.

- [77] A. Mishra, K. Alahari, and C. V. Jawahar. Scene text recognition using higher order language priors. In *BMVC*, 2012.
- [78] A. Mishra, K. Alahari, and C. V. Jawahar. Top-down and bottom-up cues for scene text recognition. In *CVPR*, 2012.
- [79] A. Mishra, K. Alahari, and C. V. Jawahar. Image retrieval using textual cues. In *ICCV*, 2013.
- [80] G. Nagy. Twenty years of document image analysis in pami. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000.
- [81] L. Neumann and J. Matas. A method for text localization and recognition in real-world images. In *ACCV*, 2010.
- [82] L. Neumann and J. Matas. Text localization in real-world images using efficiently pruned exhaustive search. In *ICDAR*, 2011.
- [83] L. Neumann and J. Matas. Real-time scene text localization and recognition. In *CVPR*, 2012.
- [84] L. Neumann and J. Matas. On combining multiple segmentations in scene text recognition. In *DAS*, 2013.
- [85] L. Neumann and J. Matas. Scene text localization and recognition with oriented stroke detection. In *ICCV*, 2013.
- [86] T. Novikova, O. Barinova, P. Kohli, and V. Lempitsky. Large-lexicon attribute-consistent text recognition in natural images. In *ECCV*, 2012.
- [87] L. O’Gorman. The document spectrum for page layout analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1993.
- [88] N. Otsu. A threshold selection method from gray-level histograms. *Automatica*, 1979.
- [89] U. Pal and B. Chaudhuri. Indian script character recognition: a survey. *Pattern Recognition*, 2004.
- [90] T. Q. Phan, P. Shivakumara, and C. L. Tan. A laplacian method for video text detection. In *ICDAR*, 2009.
- [91] R. Plamondon and S. N. Srihari. Online and off-line handwriting recognition: a comprehensive survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000.
- [92] P. Rantalankila, J. Kannala, and E. Rahtu. Generating object segmentation proposals using global and local search. In *CVPR*, 2014.
- [93] T. M. Rath and R. Manmatha. Word image matching using dynamic time warping. In *CVPR*, 2003.
- [94] J. Rodriguez and F. Perronnin. Label embedding for text recognition. In *BMVC*, 2013.
- [95] Y. Rui, T. S. Huang, and S.-F. Chang. Image retrieval: Current techniques, promising directions, and open issues. *Journal of Visual Communication and Image Representation*, 1999.
- [96] J. Sauvola and M. Pietikäinen. Adaptive document image binarization. *Pattern Recognition*, 2000.
- [97] A. Shahab, F. Shafait, and A. Dengel. Icdar 2011 robust reading competition challenge 2: Reading text in scene images. In *ICDAR*, 2011.
- [98] A. Shahab, F. Shafait, A. Dengel, and S. Uchida. How salient is scene text? In *DAS*, 2012.
- [99] K. Sheshadri and S. K. Divvala. Exemplar driven character recognition in the wild. In *BMVC*, 2012.

- [100] C. Shi, C. Wang, B. Xiao, Y. Zhang, and S. Gao. Scene text detection using graph model built upon maximally stable extremal regions. *Pattern Recognition Letters*, 2013.
- [101] C. Shi, C. Wang, B. Xiao, Y. Zhang, S. Gao, and Z. Zhang. Scene Text Recognition Using Part-Based Tree-Structured Character Detection. In *CVPR*, 2013.
- [102] P. Shivakumara, W. Huang, and C. L. Tan. An efficient edge based technique for text detection in video frames. In *DAS*, 2008.
- [103] P. Shivakumara, W. Huang, and C. L. Tan. Efficient video text detection using edge features. In *ICPR*, 2008.
- [104] P. Shivakumara, T. Q. Phan, and C. L. Tan. A robust wavelet transform based technique for video text detection. In *ICDAR*, 2009.
- [105] P. Shivakumara, T. Q. Phan, and C. L. Tan. Video text detection based on filters and edge features. In *International Conference on Multimedia and Expo*, 2009.
- [106] P. Shivakumara, T. Q. Phan, and C. L. Tan. New fourier-statistical features in rgb space for video text detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 2010.
- [107] P. Shivakumara, T. Q. Phan, and C. L. Tan. New wavelet and color features for text detection in video. In *ICPR*, 2010.
- [108] A. W. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000.
- [109] R. Smith. An overview of the tesseract ocr engine. In *ICDAR*, 2007.
- [110] E. Spyrou and P. Mylonas. A survey of geo-tagged multimedia content analysis within flickr. In *Artificial Intelligence Applications and Innovations*. 2014.
- [111] C. C. Tappert, C. Y. Suen, and T. Wakahara. The state of the art in online handwriting recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1990.
- [112] R. Tarjan. Depth-first search and linear graph algorithms. *SIAM Journal on Computing*, 1972.
- [113] S. Tian, S. Lu, B. Su, and C. L. Tan. Scene text recognition using co-occurrence of histogram of oriented gradients. In *ICDAR*, 2013.
- [114] J. Uijlings, K. van de Sande, T. Gevers, and A. Smeulders. Selective search for object recognition. *International Journal of Computer Vision*, 2013.
- [115] R. C. Veltkamp and M. Tanase. Content-based image retrieval systems: A survey. 2001.
- [116] K. Wang, B. Babenko, and S. Belongie. End-to-end scene text recognition. In *ICCV*, 2011.
- [117] K. Wang, B. Babenko, and S. Belongie. End-to-End Scene Text Recognition. In *ICCV*, 2011.
- [118] K. Wang and S. Belongie. Word Spotting in the Wild. In *ECCV*, 2010.
- [119] Q. Wang, T. Xia, L. Li, and C. L. Tan. Document image enhancement using directional wavelet. In *CVPR*, 2003.
- [120] J. Weinman, Z. Butler, D. Knoll, and J. Feild. Toward Integrated Scene Text Reading. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014.

- [121] J. J. Weinman, E. Learned-Miller, and A. R. Hanson. Scene Text Recognition Using Similarity and a Lexicon with Sparse Belief Propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2009.
- [122] I. H. Witten, A. Moffat, and T. C. Bell. *Managing gigabytes: compressing and indexing documents and images*. 1999.
- [123] C. Wolf and J.-M. Jolion. Extraction and recognition of artificial text in multimedia documents. *Formal Pattern Analysis & Applications*, 2004.
- [124] C. Wolf and J.-M. Jolion. Object count/area graphs for the evaluation of object detection and segmentation algorithms. *International Journal of Document Analysis and Recognition*, 2006.
- [125] C. Wolf, J.-M. Jolion, and F. Chassaing. Text localization, enhancement and binarization in multimedia documents. In *Pattern Recognition*, 2002.
- [126] V. Wu, R. Manmatha, and E. M. Riseman. Textfinder: An automatic system to detect and recognize text in images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1999.
- [127] J. Xu and W. B. Croft. Query expansion using local and global document analysis. In *ACM SIGIR*, 1996.
- [128] H.-W. H. Xu-Cheng Yin, Xuwang Yin and K. Huang. Robust text detection in natural scene images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014.
- [129] C. Yao, X. Bai, and W. Liu. A unified framework for multioriented text detection and recognition. *IEEE Transactions on Image Processing*, 2014.
- [130] C. Yao, X. Bai, W. Liu, Y. Ma, and Z. Tu. Detecting texts of arbitrary orientations in natural images. In *CVPR*, 2012.
- [131] Q. Ye, Q. Huang, W. Gao, and D. Zhao. Fast and robust text detection in images and video frames. *Image and Vision Computing*, 2005.
- [132] X.-C. Yin, X. Yin, K. Huang, and H.-W. Hao. Accurate and robust text detection: a step-in for text retrieval in natural scene images. In *ACM SIGIR*, 2013.
- [133] H. Zhang, K. Zhao, Y.-Z. Song, and J. Guo. Text extraction from natural scene image: A survey. *Neuro-computing*, 2013.
- [134] J. Zhang and R. Kasturi. Extraction of text objects in video documents: Recent progress. In *DAS*, 2008.
- [135] Y. Zhong, H. Zhang, and A. K. Jain. Automatic caption localization in compressed video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000.
- [136] Y. Zhu, C. Yao, and X. Bai. Scene text detection and recognition: Recent advances and future trends. *Frontiers of Computer Science*, 2015.
- [137] C. L. Zitnick and P. Dollar. Edge boxes: Locating object proposals from edges. In *ECCV*, 2014.