College of Computer and Information Sciences
Computer Science Department

"Text detection and recognition in machine learning"

CSC 361: Artificial Intelligence

Prepared by:

Students names

Second Semester 2018/2019

I. INTRODUCTION

Looking around us, we realize how this generation has been more digital driven. Considering the age differences, and mobility in the surroundings, using technology is affecting wide range of fields. It has been noticed that each group has its own purposes and drivers of implementing whatever makes their lives easier and solves difficulties that may arise during daily activities. One of the trending technologies that is spreading nowadays is the uses of artificial intelligence techniques and machine learning algorithms in text recognition.

Text detection has been defined as "the process of detecting and locating those regions that contain texts from a given image and is the first step in obtaining textual information"[1] as shown in Figure1. Text detection and recognition can be very essential in many schemes. It can be beneficial to those who travel or live among people who speak and use different languages. It can be used for translating foreign letters and signage instantly. It can be very helpful in many applications used for elderly people and those whom are visually impaired. Moreover, text recognition is a class of computer vision problems along with many other problems including optical character recognition (OCR) and text detection.

Text detection and recognition both have been noticed as a very important problem recently, which got a lot of researchers' attention. Even though a considerable amount of work has been done in both fields text detection and recognition, it is still a very challenging problem. There are different types of images containing text, such as document images, scene images. Scene images are usually captured by digital cameras and smart phones which implement acquisition of images and videos containing scene text such as shop signs on the streets, advertisements, and restaurant menus. Such devices also bring up new image processing problems for instance sensor noise, viewing orientation, contrast, brightness, and poor image resolutions. As a result of the mentioned problems many different techniques are being applied for different types of images. In this paper, we will give a literature review of the many available techniques and some of their strengths and weaknesses. Primarily from the point of view of representation.

Figure 1: Illustration of scene text detection and recognition.

II. BACKGROUND

A. Neural networks

Artificial neural networks are inspired from the humans' biological neural network. It combines many different machine learning algorithms that work together so it can process different

and complex data or input. Neural networks are made to detect patterns. The Neural network input is described through a kind of machine recognition. It recognizes numerical patterns that are represented as vectors which contain real-world data like images, text and sounds.
Artificial neural networks help us in classifying and clustering data. It is similar to a classify layer on the top of the data. It helps in classifying and grouping the similarity among data. Then they classify the data corresponding to a labeled dataset to train on.

B. support vector machine

Support Vector Machines (SVM) are supervised learning models that analyze data that are used for classifying data using learning algorithms. SVM is suitable for extreme cases, it takes the input or data and classify it into two categories. Then it separates the data into two classes of data points using hyperplanes which decide the boundaries of classifying. The support vectors are the data points that are close to the hyperplane. Using these support vectors, we maximize the margin of the classification as shown in the figure 2.

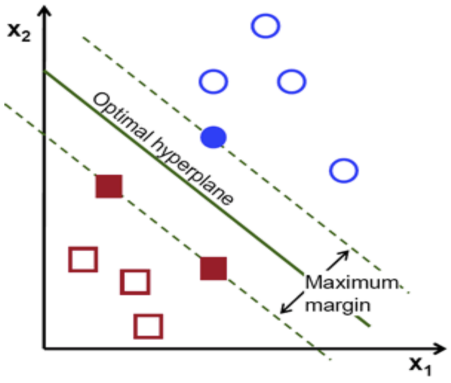

Figure 2: the maximum margin separator.

C. Clustering

Clustering is the process of dividing the dataset into multiple groups that are called clusters. Each cluster contains a subset of the dataset, each subset in a cluster shares the same or similar features and is different from other clusters. In other words, the goal of clustering is to isolate groups with the same or similar features and assign them into clusters. Clustering is considered an unsupervised learning method, as the labels for the data are unknown and no previously made classes are used to make the clusters. There are many types of clustering algorithms such as Hierarchical clustering, Fuzzy clustering, Density-based clustering and the famously known K-means clustering algorithm.

III. TEXT DETECTION

There are many ways to detect texts in images and videos, but there are three main types: texture-based method, region-based method and hybrid method.

A. Texture based method:

in this method they try to process text as special type and try to use their advantages. To differentiate between text and non-text area in videos and images we have multiple methods such as local intensities, wavelet coefficients and filter responses. they are very useful but in other way they are expensive since all scene and locations should be scanned.

Wang et al.[2] has proposed a method that will be fed with all possible types of input from the city scene such as shops signs, street directions and names. then they used AdaBoost algorithm to classify the signs and pictures in the databases so they can enable the algorithm to train and evaluate the algorithm efficiency. First, we detect a single character using a sliding window. Then they select the feature set as guided by the principle of *informative features mentioned in* [3] *and they calculate the feature set* joint probability distributions to obtain the weak classifier as log-likelihood ratio tests. we apply sub-regions of the image and outputs text candidate regions to the strong classifier. Then they used binarization algorithm that takes the text region from strong classifier as input and extends the region to make sure that regions that did not get covered by the strong classifier are included. At the end ORC software determine whether are text and read them or reject them.

Wang et al. method runs so fast, but it only detects the that given in the list. In 2013 Kim et al. [1] proposed a system that use support vector machines (SVM) to classify text without need of previous knowledge. rather than use texture feature extraction module it fed the intensities of the raw pixels directly with (SVM). After (SVM) each pixel represents Its probability in the input image of being part of text region. They use continuously adaptive mean shift algorithm (CAMSHIFT) to identify the text region. The combine of SVM and CAMSHIFT gives a powerful image and Video detection and work for all dimensions (Figure. 2). Kim et al. method work perfectly when we use a single-colored image or video but in multi-colored images it not the most efficient method to use.

Figure 3: Text detection examples of t Kim et al algorithm. It shows a representative work in a simple Video scene
Source: From [1].

B. Connected component method:

At first, the image is divided then extract candidate components through many ways. Then Using some rules, we filter the non-text components. connected component method is much more efficient because the number of processed components is approximately small.

in the method proposed by Epshtein et al.[4] it that use stroke width transform which is a new image operator. stroke width transform is a significant feature to distinguish the text from edge

maps and calculating the stroke width in each pixel in the image. After that they use Finding letter candidates by grouping two neighbor pixels to see if they have similar stroke width.
Figure 4 has described how Epshtein et al.[4] algorithm works.

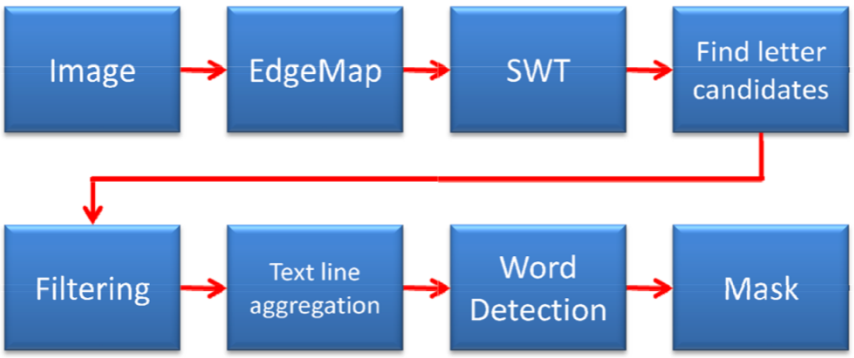

Figure. 4: the flow chart of Epshtein et al. algorithm. Source: From [4].

Epshtein et al. [4] algorithm works perfectly with different fonts and languages. But to work with complex images Yi et al.[5] proposed an algorithm that work efficiently with complex images and backgrounds. First, they used image partition to group pixels that belongs to the same character and obtain binary map of candidate character components. Then we detect text by grouping character candidate based on joint structural features such as character sizes, distance between character…etc. they put an assumption that every text string has at least three characters. They propose two algorithms to detect text string adjacent character grouping method, text line grouping method. In adjacent character grouping method, it calculates the sibling groups for every character candidate as string segments and merges the intersecting sibling into text string.
Yi et al.[5] method performs better than. Epshtein et al.[4] since it works for non-horizontal orientations.

C. Hybrid methods:

It is a combination of the benefits text-based method and connected component method. The hybrid algorithm proposed by Liu et al.[6] use clarify color image edge detection algorithm to extract all the pixels in text edge. Then they used connected component method to detect the external outline on the edge pixels. They scan all region outline to construct the candidate text regions and classify part non-text regions. For every candidate text region is confirmed with texture features derived from wavelet domain. In the end, the expectation maximization algorithm is used to write text region in binary code to prepare data for recognition. It works efficiently with different background colors, languages and different character sizes.

In the other hand, pan et al. [7] used hybrid approach to strongly localize and detect texts in images. The detection of text region is designed to evaluate the text existing, to filter the non-text components a conditional random field (CRF) model that looks for unary component properties and binary contextual component relationships with supervised parameter learning is recommended. then text components are grouped into lines with a learning-based energy

minimization method. Sometimes this approach fails in detecting text that have complex images behind.

IV. TEXT RECOGNITION:

There are many classifications of Text Recognition approaches, but in this section, we are going to classify them based on the training techniques used:

A. Artificial neural networks:

There are many types of neural networks, such as radial basis function (RBF) network, multilayer perceptron (MLP), higher-order neural network (HONN), Convolutional Neural Networks (CNN), and so many more that have been applied to text and pattern recognition here are some of the approaches that used neural networks and proven to be efficient in text recognition.

In the system made T. Wang, D. Wu, A. Coates and A. Ng in [8] an unsupervised feature learning algorithm was used; the algorithm automatically removes features from given data.
For the text recognition part, Wang et al. [8] used the system in [9] as part of their method. Wang et al.[8] incorporated learned features into a large, discriminatively-trained convolutional neural network (CNN). By using the representative feature of the convolutional neural network as shown in Figure 5. they were able to train precise text detectors and character recognizing modules.

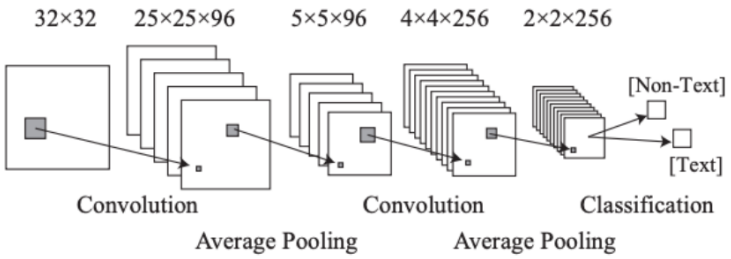

Figure 5: CNN used for text detection. Source: From [8].

First, Wang et al. [8] performed sliding window detection to classify candidate lines of text, which Wang et al.[8] later, perform word-level segmentation and recognition on, to get the end-to-end results. For both detection and recognition, Wang et al.[8] used multilayer convolutional neural network (CNN)(Figure 5). The first layer of the network is trained using the unsupervised learning algorithm in [9] which is a learning algorithm that tries to learn the necessary features straight from the data as opposed to using purpose constructed, text-definitive- models or structures. Then as it is usual in CNNs, they calculated the average pool over the first layer response map then stack another convolution and average pooled the layer on top of the first layer. They trained the network by backpropagating the L2-SVM classification error,2 then fixed the filters in the first convolution layer. The system combines a lexicon with detection and recognition modules using post processing techniques such as non-maximum suppression (NMS) [10] and beam search [11].

Basically, Wang et al.[8] pipeline works by first detecting and locating the candidate text lines then integrating the character responses using beam search [11]. Wang et al.[8] built an end-to-end system with only easy post-processing procedure such as non-maximal suppression (NMS)

[10] and beam search [11] the system achieved state-of-the-art performance and impressive results on different benchmarks.

Just like the Wang et al. in [8] Coates *et al.* in [12] built a similar system also using artificial neural networks. The framework proposed in [12] did not need any human-characterized data and performed word recognition on the complete image holistically. The deep neural network models at the center of this framework are trained merely on synthetic text (Figure 6). Coates *et al.* in [12] considered using three models, 90k-way dictionary encoding, character sequence encoding, and bag-of-N-grams encoding.

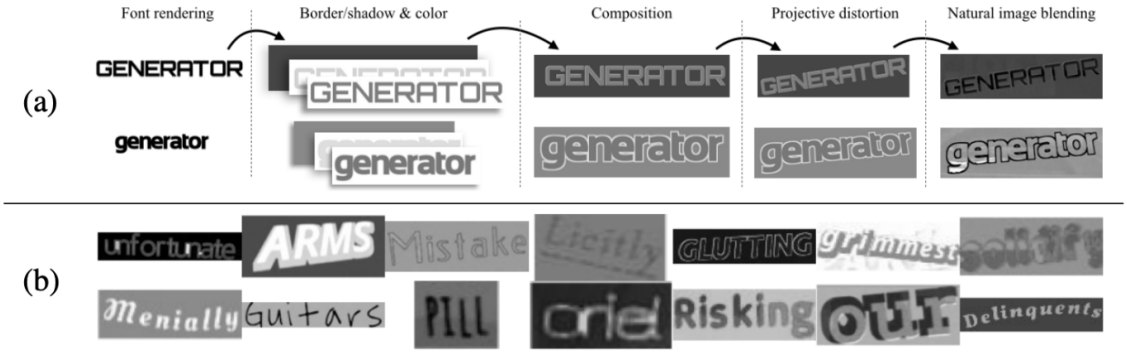

Figure 6: (a) The text generation process after font rendering, creating and coloring the image layers, applying projective distortions, and after image blending. (b) Some randomly sampled data created by the synthetic text engine. Source: From [12].

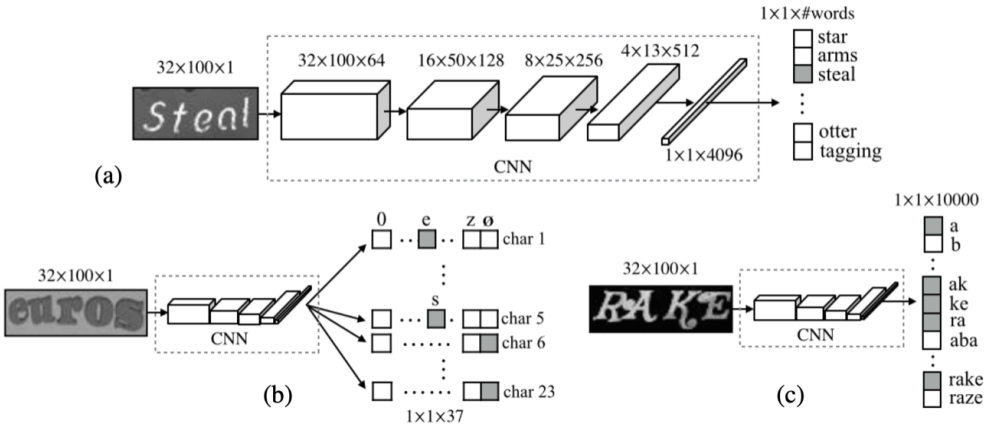

Figure 7: A schematic of the CNNs used showing the dimensions of the feature maps at each stage for (a) dictionary encoding, (b) character sequence encoding, and (c) bag-of-N-gram encoding. The same five-layer, base CNN architecture is used for all three models. Source: From [12].

a. First model, 90k-way dictionary encoding:

They used a state-of-the-art CNN (Figure 7.) text recognizer that pools marks from images of complete words. Then regress all the characters instantaneously, expressing this as a classification problem in a large lexicon of 90k probable words. The CNN is trained to identify an extremely large amount of words using incremental training. The words are designated in a predefined dictionary. For the training part they trained the network by back-propagating the standard multinomial logistic regression loss with dropout [13].

Optimization uses stochastic gradient descent (SGD), dynamically decreasing the learning ratio as training progresses.

b. Second Model, Encoding Sequences of Characters:

This model applies a single CNN (Figure 7.) containing multiple independent classifiers, each classifier calculates the prediction of the character at a given position in the word. Each character at a given position is predicted using one classifier, this is done to each character of each word. Since the length of the word is unknown at test time the length is defined as a variable with the value 23 as the it is the maximum length of a word in the training set and define a null character class.

c. Third Model, Encoding Bags of N-grams:

This model is similar to the sequential character encoding, but words can be perceived as an arrangement of an unorganized set of character N-grams, a bag-of-N-grams. The results from the connected layers are used as probabilities of an N-gram presence in the image this is achieved by using the logistic function to each neuron. The CNN (Figure 7.) is then learns to find and recognize each N-gram anywhere in an image.

For the training part they used a logistic function then back-propagated logistic regression loss with consideration to each N-gram class.

B. Kernel methods

Just as neural networks have many types, kernel methods have many types such as support vector machines (SVM), kernel perceptron, principal components analysis (PCA), canonical correlation analysis, Gaussian processes, ridge regression, spectral clustering and so many more. In this section are some of the approaches that used kernel methods in text recognition systems**:**

Support Vector Machines (SVMs) have been proven to be very successful in many areas, such as text classification. The problem with SVMs is that the basic geometric organization of text data gets overlooked by standard kernels frequently. In the system proposed by Zhang et al. in [14] they assumed that documents are presented on a multinomial manifold which is "the parameter space of the multinomial distribution"[15], that is a simplex of multinomial models supplied with the Riemannian structure produced by the Fisher information metric.

Zhang et al. [14] have proved that the Negative Geodesic Distance (NGD) on the multinomial manifold is conditionally positive definite (CPD) and used it as a kernel in their SVMs. Zhang et al.[14] proposed the idea of presenting document feature vectors as points in a Riemannian manifold, instead of the much bigger Euclidean space. Finally, they proved that the Negative Geodesic Distance (NGD) on the multinomial manifold is a conditionally positive definite (CPD) kernel, and its accuracy is better than kernels using for text classification the Euclidean geometry.

Support vector machines (SVM) has proven to be very successful in many real-world learning applications, but most machine learning algorithms are commonly applied to a randomly selected training set that has been already classified.

Tong et al. [16] introduced an algorithm that performs active learning with support vector machines, for example, algorithm for selecting which instances to request next. Tong et al. [16] allowed the learner to actively select the training data then used Pool-based active learning for classification and introduces the notion of a version space and attempted to reduce version space

as much as possible at each query. Tong et al.[16] also proved that the Simple method [16] performs the fastest computations. However, the Simple method showed less accurate results and gives unstable approximations. If each query's cost is expensive relative to computing time then Tong et al.[16] suggest using one of either the MaxMin or Ratio methods in [16]. Moreover, in the case of cheap query costs then the Simple method will work best. Tong et al. [16] have proved that the use of any of the their methods for learning will considerably exceed the standard passive learning.

C. Clustering

Clustering has also been used in many text recognition systems, in this section are some of its applications in the text recognition area:

Jamnejad et al. [17] proposed a flexible method to recognize and classify Persian texts using a thesaurus as a it part of its knowledge. As part of utilizing the thesaurus, the method has been found to recognize a more representative set of word frequencies unlike the those recognized when the method does not use the thesaurus as part of its knowledge. The relationships between words that have been used when applying the thesaurus are k-nearest neighbor classifier (KNN), decision tree classifier. The k-means clustering algorithm is applied as classifiers over the frequency-based features.

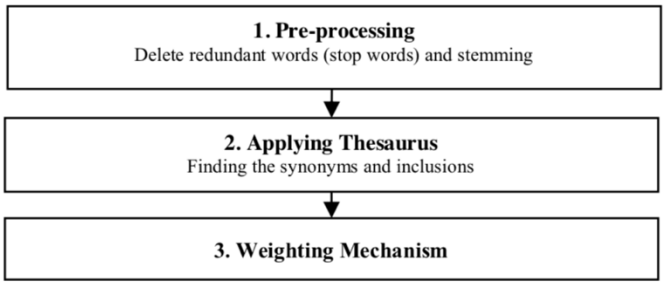

Figure 8: Proposed indexing framework.
Source: From [17]

Jamnejad et al.[17] used k-means clustering, which is a method of cluster analysis which objective is to partition the observations into k clusters, where each observation belongs to a cluster which has the nearest mean. In [17] the K- mean clustering is applied as a classifier, were Jamnejad et al. [17] assumed that the labels for the data used are known in its evaluation stage, for comparing with KNN classifier. First, in the preprocessing step, the text is filtered into valuable text to get rid of the words that are unnecessary for the process of keyword extraction. Then using Table 1. with word frequencies, they partitioned words it into two types: a) head type and b) child type. The words with head type are the only ones that are considered in the final step.

By obtaining the table of words, the weight for a synonym/antonym relationship is considered by each occurrence the synonym/antonym of a word is equal to the occurrence rate of the original word. Also, another relationship that might be considered is inclusion. The method in [17] improved the performance of Persian text classification. Jamnejad et al. [17] used a thesaurus to measure the frequencies of words. They also proved that with the use of a simple classifier, that is using a thesaurus will improve the accuracy of the classification of texts. The method considered two types of relationships a) synonyms and b) inclusion. The weighting methods used were hierarchical inclusion weighting, and linear synonym weighting.

word	frequency	Type
.	.	.
.	.	.
.	.	.
word1	3	Head
word2	3	Child
word3	3	Child
.	.	.
.	.	.
.	.	.

Table1. Table with frequencies of words.
Source: From [17]

V. DISCUSSION:

For the texture-based methods, Kim et al. algorithm [1]can detect text in natural images and videos, it works best with simple images and images/videos containing horizontal text orientation. Whereas, Wang et al. [2] algorithm is better as it can detect text in more complex natural images, but it also only works on images containing text in horizontal orientation. It also requires a lexicon for each image in order to give the needed results. Moreover, for the connected components methods, Epshtein et al. [4] algorithm also can detect text in complex natural images, but it can handle multilingual texts too. What makes it better than the previously mentioned algorithms that it is the fastest of them all. Sadly, it has the same problem of only working on images with horizontal text orientation or near horizontal text orientation. The main problem with this algorithm [4] that it requires manually defined rules to work. Another algorithm that has the same problem of requiring manually defined rules and can handle multilingual texts is Yi et al. [5] but, Yi et al. [5] algorithm can give impressive results with images that contain text in different orientations, but sadly as in every other algorithm it has its weak points. It only works on simple natural images. In hybrid method Liu et al.[6] and pan et al. [7] approaches work with many different appearance, backgrounds and images colors and all languages, but the time complexity of pan et al. [7] should be accelerate.

As for text recognition, the system made by T. Wang, D. Wu, A. Coates and A. Ng in [8] is considered robust and seems to have very encouraging future, as it can be extended to a general purpose setting by dividing the conventional open-source spell checkers. Similarly, the system made by M. Jaderberg, K. Simonyan, A. Vedaldi and A. Zisserman in [12] used convolutional

neural networks, but Jaderberg et al. [12] used a synthetic dataset, which is a data set generated by mixing different fonts with different sizes and adding distortions. The synthetic dataset was used instead of real-world data, and their labels were generated from a chosen lexicon. By creating and training a system with datasets that are much bigger than what has been used in other systems, Jaderberg et al. [12] were able to apply data-hungry deep learning algorithms to train better, whole-word-based system models. The framework in [12] can be extended to much bigger vocabularies and many other languages without the need for any human labelling, which makes this framework easier to improve and develop. On the other hand, systems that used Kernel methods for training, such as the ones mentioned in [14], [16], whereas in [14] the algorithms were made for performing active learning with SVMs by taking advantage of the combination between parameter space and feature space. Tong et al.[16] introduced three new algorithms that try to reduce version space for each query. Using such algorithms can improve both the transductive and inductive settings. The system proposed in [14] can be extended to Negative Geodesic Distance (NGD) kernel, and combine it with other kernel methods for pattern analysis systems such as, kernel principal component analysis (PCA). The case for systems that used clustering algorithms for their training process as done in [17] where their proposed method improved the performance of text classification. The method used a thesaurus to support the process of finding words frequencies. Jamnejad et al. in [17] proved that both clustering and text classification showed remarkable results when a thesaurus is used. New weighting methods can be used to improve the efficiency and accuracy of this method.

VI. CONCLUSION:

Despite the drastic changes ongoing nowadays in the field of text recognition, and all the new methods that are changing rapidly, one cannot give a definitive answer to the all-time question of which method/approach is the best? we concluded that the answer for this question is not easy to find in the field of text recognition. Choosing a method will always depend on the purpose of its use and the type of its input/data for instance, processing handwritten text will require different classification methods than texts from natural scenes. Each method has its weak and strength points. When deciding what method to use, an analysis of the problem must be made to make the appropriate decision.

VII. REFERENCES

[1] 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 Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 12, pp. 1631–1639, 2003.

[2] K. Wang and S. Belongie, “Word Spotting in the Wild.pdf,” vol. 6311, pp. 591–604, 2011.

[3] P. Viola and M. Jones, “Fast and Robust Classification using Asymmetric AdaBoost and a Detector Cascade,” *NIPS*, p. 1311–1318., 2001.

[4] B. Epshtein, E. Ofek, and Y. Wexler, “Detecting Text in Natural Scenes with Stroke Width Transform,” *CVPR*, pp. 2963–2970, 2010.

[5] Chucai Yi and YingLi Tian, “Text String Detection From Natural Scenes by Structure-

Based Partition and Grouping," *IEEE Trans. Image Process.*, vol. 20, no. 9, pp. 2594–2605, 2011.
[6] Y. LIU, S. GOTO, and T. IKENAGA, "A Contour-Based Robust Algorithm for Text Detection in Color Images," *IEICE Trans.*, vol. E89–D, no. 3, pp. 1221–1230, 2006.
[7] Y. Pan, X. Hou, and C. Liu, "A Hybrid Approach to Detect and Localize Texts in Natural Scene Images," *IEEE Trans. Image Process.*, vol. 20, no. 3, pp. 800–813, 2011.
[8] T. Wang, D. J. Wu, A. Coates, and A. Y. Ng, "End-to-End Text Recognition with Convolutional Neural Networks," *ICPR*, p. 3304–3308., 2012.
[9] A. Coates *et al.*, "Text Detection and Character Recognition in Scene Images with Unsupervised Feature Learning," *ICDAR*, p. 440–445., 2011.
[10] A. Neubeck and L. Van Gool, "Efficient Non-Maximum Suppression," *Proc. 18th Int. Conf. Pattern Recognit.*, vol. 3, pp. 850–855, 2006.
[11] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. 1995.
[12] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Synthetic Data and Artificial Neural Networks for Natural Scene Text Recognition," *CoRR*, vol. abs/1, pp. 1–10, 2014.
[13] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *CoRR,* vol. abs/1207.0, pp. 1–18, 2012.
[14] D. Zhang, X. Chen, and W. S. Lee, "Text Classification with Kernels on the Multinomial Manifold," *SIGIR*, p. 266–273., 2005.
[15] G. Lebanon and J. Lafferty, "Hyperplane Margin Classifiers on the Multinomial Manifold," *ICML*, vol. 69, 2004.
[16] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," *J. Mach. Learn. Res.*, vol. 3, pp. 45–66, 2001.
[17] M. I. Jamnejad, A. Heidarzadegan, and M. Meshki, "Text Recognition with k-means Clustering," *Res. Comput. Sci.*, vol. 84, pp. 29–40, 2014.