REPUBLIQUE ALGERIENNE DEMOCRATIQUE & POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR & DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE DJILLALI LIABES
FACULTE DES SCIENCES
SIDI BEL-ABBÈS

BP 89 SBA 22000 –ALGERIE- TEL/FAX 048-54-43-44

THESE

Présentée par : OULLADJI LATEFA

Pour obtenir le Diplôme de Doctorat Sciences
Spécialité : Informatique
Option : Intelligence Artificielle

Intitulé

Détection du Texte Arabe En Utilisant Les Méthodes Statistiques et d'Apprentissage Automatique

Président	Dr. Boukli Hacene Sofiane	MCA à l'UDL-SBA
Directeur de thèse	Pr. Batouche Mohamed	Professeur à Constantine 2/ Constantine
C0-Directeur de thèse	Pr. Faroun Mohamed Kamel	Professeur à l'UDL-SBA

Examinateurs

Dr. TOUMOUH Adil	MCA à l'UDL-SBA
Pr. RAHMOUN Abdellatif	Professeur à L'ESI-SBA
Dr. AMAR BENSABER Djamel	MCA à l'ESI-SBA

Abstract

The automatic detection and recognition of zone text in natural images remain indispensable due to the omnipresent of text information in daily human life. This domain contoured a development of many applications specially with English language where many systems were implemented and proved their efficiency. Arabic language represents a real challenge for its cursive nature and rich vocabulary. The first step of our work was inspired from Gomez and Karatzas [17] on multiscript detection using Gestalt theory. For the second step, we implemented three classifiers namely Neural Network, Support Vector machine and Adaboost. These classifiers were deployed to classify the group regions in images as text or non-text. To improve the system performance an ensemble method based on majority voting was applied where the outputs of the three classifiers were fused. Experiments were conducted using own image database and ground-truth and the empirical results illustrate that the proposed method is efficient.

Résumé

La détection et la reconnaissance automatique des zones de texte dans des images naturelles est indispensables en raison de l'omniprésence de l'information textuelle dans la vie quotidienne des êtres humains. Ce domaine a contourné un développement de nombreuses applications spécialement avec la langue anglaise où de nombreux systèmes ont été mis en œuvre et ont prouvé leur efficacité. La langue arabe représente un véritable défi pour sa nature cursive et son riche vocabulaire. La première étape de notre travail a été inspirée du travail de Gomez et Karatzas [17] sur la détection multiscript en utilisant la théorie de Gestalt. Pour la deuxième étape, nous avons implémenté trois classifieurs : Neural Network, Support Vector Machine et Adaboost. Ces classifieurs ont été déployés pour classer les régions groupées par la premiere étape en texte ou non-texte. Pour améliorer les performances des systèmes, une méthode d'ensemble basée sur le vote de la majorité a été appliquée sur les résultats des trois classifieurs. Les expériences ont été menées en utilisant notre propre base de données d'images où les résultats empiriques illustrent que la méthode proposée est efficace.

ملخص

الاكتشاف و التعرف الآلي للمنطقة النصية في الصور الطبيعية يعتبر أساسي بسبب وجود الدائم للمعلومات النصية في الحياة اليومية للإنسان. هذا المجال يعرف تطورا ملحوظا في العديد من التطبيقات الذكية خاصة مع اللغة الانجليزية حيث تم تطوير العديد من الأنظمة التي اتبثت كفاءتها. أما بالنسبة للغة العربية فهي تمثل تحدي حقيقي بسبب طبيعتها المتداخلة في الكتابة ومفرداتها الغنية. في عملنا هذا تم استلهام الخطوة الأولى من عمل لويس[١٧] لاكتشاف المناطق المتشابهة والممكن تواجد النص فيها وفي الخطوة الثانية تم استخدام ثلاث مصنفات وهي الشبكات العصبية الاصطناعية والخورزميات الخطية وادبست. هذه المصنفات تم استعمالها لتصنيف المناطق النصية من الغير نصية في الصور الطبيعة. لتحسين اداء النظام تم دمج مخرجات النتائج الثلاث الخاصة بالمصنفات و تجميعها في واحد وذلك استنادا الى اعلى نسبة من المخرجات . تم تقييم نتائج التطبيقات باستخدام قاعدة بيانات خاصة بنا حيث اثبتث طريقة التجميع فعاليته.

Dedication

By the name of Allah, Most Gracious, Most Merciful

To My GOD

To the soul of my Father

To my dear Mother

To my wonderful children: Nasser and Leen

To my Friends: Karima, Naima, Fatima, Fouzia, Iness , Lila, Khalida, Rachida, Houda, Farah, Ibtisam. . . .etc. To my brothers and sister

Acknowledgement

I would like to offer my sincere thanks to my supervisors Feraoun Kamel and Batouche Mohammed, for their help, encouragement and guidance to make this work see the light.

My sincere gratitude goes to Professor Ajith Abraham for his encouragement, patience, guidance caring, motivation and his valuable help to accomplish both my experimental results and article with his great background in machine learning approaches. Also my deep gratitude goes for Lluis Gomez for his valuable help by providing all necessary documents and guidance to construct my database which were very helpful in achieving my objectives in research. May God recompense them.

I would Like also to thank Dr Ben-yahia Karima, Dr Boukabrine Fouzia, Dr Adjouj Reda for their valuable help and encouragement.

I would like to express my sincere thanks to my thesis committee members for their critical thoughts, insightful comments and expert advices on my dissertation. Many thanks to several people who in one way or another contributed to the completion of this thesis by their assistance and guidance.

Contents

List of Tables

List of Figures

INTRODUCTION

In recent years, scene text detection and recognition became a very important need due to daily presence of textual information's all around us and increasing technologies specially in mobile phone with high resolution camera. The availability of such application can help blind person to know where he is and where he can go for example. Also, can help foreign visitors to have a real time translation in his native language. In automatic car driving such application can assist driver by reading road panel and signs...etc. Indeed, as human to understand text we must pass by reading process that use our visual system eyes with the combination of visual cortex in our brain, where text is interpreted in easy way. This process of easy reading and understanding was a consequence of long learning days acquired from childhood. Therefore, imitation of such visual system by computer is computationally complex and challenging task despite of some nice improvement in English languages detection and recognition applications. In contrast, many other languages like Arabic language meet difficulties to develop such applications where it main problems are the rich vocabulary and cursive nature. So, this cursively in Arabic languages writing where characters are connected constituting a word or a sub-word in the language, make the segmentation of text into set of characters very difficult even in specific kind of images with restricted conditions and without background. In general, text detection, as a first important part before recognition has been treated by many researchers this last years specially in Latin script languages. Many great works had been done with remarquable effort and success to develop text detection application but still some problems remain always due to the text nature in the wild that pose always problem with its different size, color, orientation, intensity, font...etc.

For Arabic language detection not much works have been done for the detection stage due to its intricacies. Therefore, the goal of this thesis is to address this problem of Arabic language detection using Gestalt theory and machine leaning techniques. Our proposed systems are based on Gomez and Karatzas [17] works inspired by Perceptual Organization (Gestalt theory) where the grouping of meaningful regions is based on set of proximities and similarities laws. A features set of meaningful group regions is calculated to feed three separately machine learning classifiers (support vector machine (SVM), Neural Network (NN), Adaptive boosting (Adaboost)) to decide which regions are more probable to be text. Also an ensemble method based on majority voting are applied on three outputs classifiers for better decision on text region classification. For developing and evaluating such systems we construct our own database combined with ground truth as important step to our system. This latter was essential due to the big lack of existing of database containing Arabic script in natural image for public use.

In our applications we are focusing to determine regions where text exist and not localized with bounding boxes. An evaluation metrics consisting in precision, recall and accuracy were used to calculate our systems performance then, a comparative study was done between the ensemble method and each three classifiers separately, where ensemble method is found to be the best in precision and accuracy.

The thesis is organized in four chapters. In the first one we analyze the literature using three basic concepts in text detection system, the region-based, the texture-based and the hybrid-based systems with a comparative study done on their performance also on the most used databased. The second one resume available Arabic databases with a description of our own database and it evaluation protocol. In the chapter three we present a theories view of our preprocessing step with machine learning used approaches . At the end we present in chapter four the experimental results on our database and KIAST database as segmentation test of our four systems.

Chapter 1

LITERATURE REVIEW

In this chapter we present literature review over important works that underline our research subject. We start with description of three based aspects of text detection and we present some important work done in Arabic script finally we conclude with comparison.

1.1 Introduction:

Text detection in natural images is in general the basic part before recognition process. A well defined detection method can be very efficient for a powerful recognition steps. Detection method is an open research domain due to its difficulties that met great success these last years essential for detection of Latin languages'. The text zones in image processing field are defined as repetitive object with similar and specific features that distingue them from other repeated patterns and clutters like leaves, barriers, etc. The process of detection and recognition is very complex specially in the case of natural images where they are subject to many effects, first from the nature of text itself where it is present in different size, font, style, position, orientation and color. Second from the outside effect like sun light that can blur the text or presence of complex background also we can find the panel or sign that contain text that can be destroyed, deformed or have many reflecting sides. These effects

have a direct impact in intensity distribution of foreground and background of text for detection or recognition [1]. Different approaches were developed to deal with text detection where we can find some academic papers that analyze and classify those approaches using different aspect and following theirs points of interest. In general, most of the works like of Zhao et al [2] and Grond [3] divided the detection in three aspects, connected component based, texture based and hybrid based methods. Another aspect of division used by Yousfi [4] based on machine learning detection, heuristic detection and hybrid one. Following our studies, the literature review is divided in three categories as follows:

- Region-based detection
- Texture-based detection.
- Hybrid-detection.

1.2 Region-based detection:

In general text detection process aim to determine the existence or not of text in images. This based region process use bottom-up approaches starting by evolving all regions in images from low level pixel representation using edge detection, connected component or clustering candidate region. These detected regions represent characters grouped together to form probable text that can be filtered using different kind of heuristics or machine learning methods applied on complex calculated set of features. These features set characterize the text from background and others regions using color intensity, similarity, edges and spatial layout...etc. [5]. In the following we will present in details the main works done by the regions based method and brief description for the other ones.

One of the most interesting work that use region based detection with heuristic rules to extract text region is a state of the art works that belong to Epshtein et al [6] where they presented a fast novel image operator that give the value of stroke width in image. Their approach proved robustness and simplicity to detect texts with

different scale, direction and fonts of different languages. They called their operator the Stroke Width Transform (SWT), because it transforms the image data from color values to the most likely stroke described in Figure 1.1.

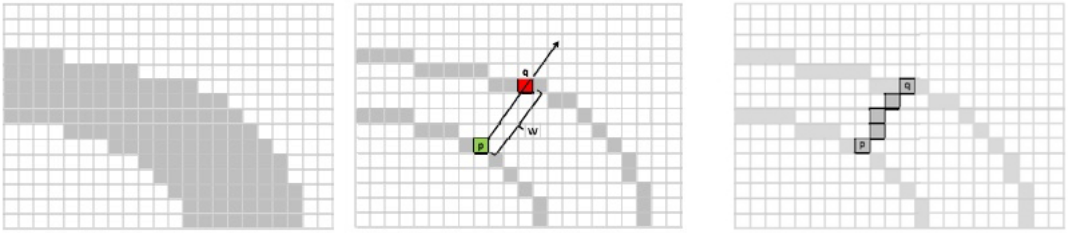

Figure 1.1: first one show typical stroke ; second one show searching direction of gradient from pixel p belonging to boundary to pixel q; the last one show the connected pixels that constitute the width.

In the first stage of their system they applied canny edge detector [7] on image then they extracted for each pixels its gradient direction. They assumed that if the pixel belong to the stroke boundary then its direction must be quit perpendicular to the orientation of stroke, basing on that assumption they calculated a ray from both pixel and its orientation to follow the probable opposite pixel in the second extremity of the stroke, if such pixel (opposite pixel to initial one) existed the SWT assign the stroke width as distance between the initial pixel and its opposite for each pixels within the segment. If those pixels had already another value of stroke width then the minimum one will be considered, otherwise if no opposite pixel was found the ray must be discarded. The next step of their approach was grouping the pixels into letter candidates where they modified the association rules in the classical connected component (CC) [8] algorithm by using a predicate that compares the SWT values of the pixel to group all neighborhood pixels that had a stroke width ratio less or equal 3. The second stage of their approach was using a set of rules on CC (grouped pixels) to extract text region by a simple calculation on stroke width variance in same CC. If there are few variations, then the CC may considered as text. They add some restriction for better result like setting a limit for the ratio between the diameter of the CC and its median stroke to be less than 10 to prevent long and narrow component. They also constrained their approach by

eliminating the two small/large components and the CC like frame and sign. Finally, they separated the lines of detected texts using horizontal histogram projection.

Using the same notion of SWT, Yao et al [9] presented both a system for detecting text in arbitrary orientation using two level classification and a database composed with images that contain different text orientation called MSRA Text Detection 500. They associated with their database a protocol that allow for a community researchers to evaluate the performance of their system and to compare their results with other methods.

Their method was assembled from approaches that use bottom-up for grouping text and top-down for chain analysis, They started as first stage by component extraction using SWT operator [6] applied on edged image which compute per pixel the width of the most likely stroke containing pixel, and then groupe the neighboring pixels that had a nearby stroke width to form a CC. The second stage was classification of these CC as text or non-text using a two-layer filtering mechanism. The first layer exploit the geometrical and statistical proprieties of CC such as the number of foreground pixels, bounding box, width and height, mean and standard deviation to classify the CC as text or non-text. The second layer was trained to identify and reject the non-text component that was hard to remove by the first one. The features of the second classifier was calculated from certain characteristic such as scale invariance and rotation invariance with the estimation of center characteristic scale and major orientation of each component. These characteristics are invariant to rotation in some degree scale and translation. A random forest was generated from the classifier where each component had a certain probability in the tree, the CCs with probabilities values lower than certain threshold were eliminated and the rest were considered as character candidates.

In the third stage, they agglomerated using a greedy hierarchical clustering [10] the character's candidate generated by the last step respecting their similarity features. The pairs of character candidates were merged recursively if they had

Figure 1.2: (a) Original image; (b) Edge detection; (c) SWT; (e) Component filtering; (f) Interpretation; (i) Detected texts.

proximal size, color and orientation. Therefore, a classifier was deployed using chain level features where elevens features were created to better classify the chain as text or non- text region. Finally, for each detected text, its orientation was calculated and its area rectangle was estimated in Figure 1.2 different process are shown. Their algorithm was performed two times with the gradient direction and with inverse direction image where the results was merged to make better decision.

Another interesting work region based approaches combined with machine learning classification was introduced by Neuman and Matas [11] [12] where they developed an Extremal Regions (ERs) algorithm based on inclusion relation for text detection in natural image. Their system had two major stages, the first one aim to achieve a real time performance during the detection of ERs. These ERs were constructed using the inclusion relation, where they were evolved sequentially respecting an incremental threshold in level-color intensity interval [0,...,255]. Each ERs in a given incremental threshold depend on its predecessors. They used in their system an incremental computable descriptors calculated in O(1) [13] during the ERs sequential grouping showed in Figure 1.3, these descriptors were Euler number, perimeter and horizontal crossing. In same times, They deployed a sequential classifier AdaBoost with decision

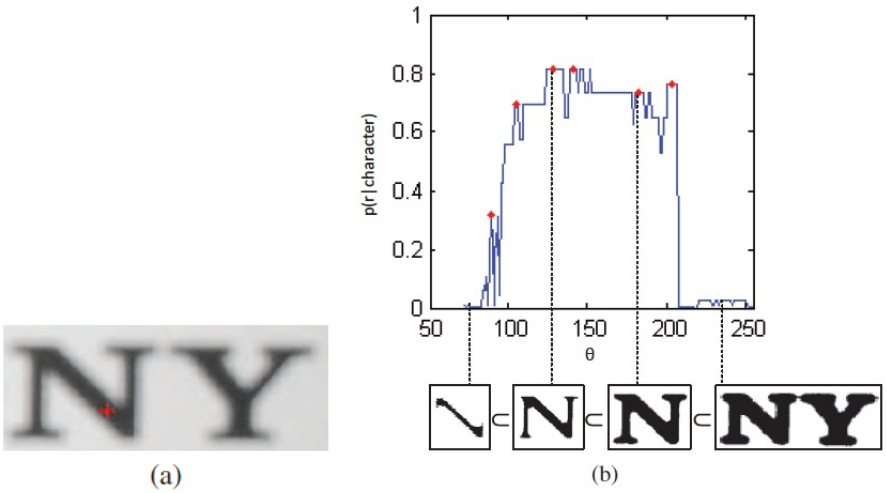

Figure 1.3: (a) A source of cutted images with initial seed of the ER; (b) p(r|character) estimated incrementally computable values in the inclusion sequences.

trees that had as input a set of features (aspect ratio, compactness, number of holes, horizontal crossings) calculated from a cited descriptors. The output of classifier was calibrated using logistic correction to assign a certain probability to each region being a character, only the region with local maximal probability was selected for the next stage of classification. In the second stage a set of more complex features was calculated with feedback loop in an exhaustive search applied to group of ERs selected by first classifier. They used a Support Vector Machines (SVM) classifier with RBF Kernel [14] to recognize the ERs as character or non-character, the SVM was trained using features of the first classifier plus others more computationally expensive like hole ratio, convex hole ratio and the number of outer boundary inflexion points.

A particular case of ER called MSER was introduced by Yin et al [1] as state of the art. Their work was based on method that use a Maximally Stable Extremal Regions (MSER) [15] for scene text detection. The first stage of their method was character candidates extraction by applying a MSERS algorithm on input text image to generate a tree that contain all different image regions. A parent/children elimination was performed in the resulting tree to prune a non-character's candidates. The choice of who can be a character, child or parent was estimated by their

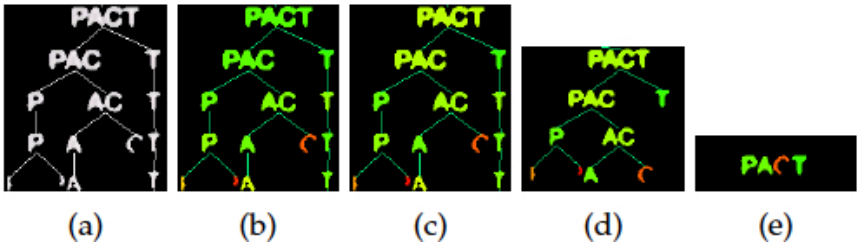

Figure 1.4: (a) MSER tree of a text segment; (b) MSERs colored following the variations (c) regularized variations; (d) linear reduction; (e) character candidates after tree accumulation.

proposed regularization variation (RV) scheme where two algorithms for elimination was deployed, called respectively linear reduction and tree accumulation algorithm. The RV was achieved by penalizing the variation of regions with maximal and minimal aspect ratios, next a linear reduction algorithm was applied to resulting tree with the new RV values where in the case of parent that had only one child, the reducing algorithm choose the one who had minimum variation and discard the second. Finally, as last step in this stage, they used a tree accumulation algorithm for the case of parent that had more than one child as Figure 1.4 show. If one of the child's had lower variation than parent, the algorithm kept them, otherwise they discarded them from tree. The advantages of both algorithms was in recovering the lack of the ordinary pruning algorithm that had limitation in choosing the child as maximal region also the both algorithms present a linear complexity in calculation.

The second stage of their method was text candidates construction using a single link-clustering algorithm [16]. This algorithm had as input a set of started points representing the probable character candidates resulting from the last stage and had as output a set of clusters that correspond to the text candidates, these clusters were grouped iteratively using a weighted features distance function that compute the degree of similarity between data points. The features space of distance function used in their algorithm was a vector composed from the characteristics of probable character candidates, like width and height differences, top and bottom alignments, color and stroke width difference. The weighted values of features and threshold were

learned automatically using distance metric learning algorithm.

The final stage of their method was text candidates elimination. For each text candidate a posterior probability was measured using classifier where high probability corresponds to a non-text region to be removed. The text-candidate features were inspired from the uniformity of characters feature in the text and calculated as input for the classifier. These features were smoothness (average difference of adjacent boundary pixels' gradient directions), stroke width, average stroke width, stroke width variation, height and width with aspect ratio. The following Figure 1.5 show the flowchart of their system associated with the experimental results.

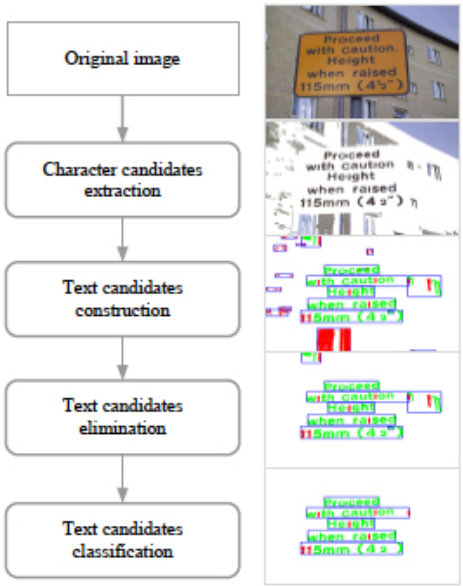

Figure 1.5: Flowchart of the system and the associated experimental results.

Gomez and Karatzas [17] added gestalt theory [18] that compute the perceptual organization of system through various laws of similarity combining with proximity laws to have the evidence on the most meaningful groups of MSER. The pipeline of Gomez approach for text extraction was composed from three stages. The first stage consists on detecting text character using region decomposition algorithm MSER. In

the second stage they applied in three steps the perceptual organization clustering notion on resulting tree where in the first step, they combined the geometrical and statistical features of the most meaningful groups of regions with the coordinates of the regions centers to describe similarity and proximity relations between characters of a word or text line. For each feature sub-spaces, a single linkage clustering analysis was performed to build a set of nodes that constitute a dendrograms. The second step of that stage consisted on testing the meaningfulness of dendrograms nodes using Helmholtz principle as probabilistic approach to Gestalt Theory by automatically detecting deviation from randomness corresponding to meaningful events. To assess the meaningfulness of grouped regions they calculated a metric for each node with merged regions in dendrograms using binomial distribution to know if the observed distribution come by chance or not, the low value the metric had the most region was meaningfulness. Then they analyzed the cluster of meaningful nodes by comparing the values of their metrics at each node in the dendrogram. The last step consists on accumulation by evidence [19] all maximum meaningfulness clusters of each dendrograms using a co-occurrence matrix to perform the final clustering analysis of the regions. Their notion of meaningfulness resulted in all kind of similarities detected in the images for that it was necessary for them to filter the result of the above stage and keep only the meaningfulness clusters considered as text. In their final stage they used a combination of two Real Adaboost classifiers trained using different scripts with ICDAR2003 and MSRA-TD500[9] data sets. The first classifier was used to classify each region in each group being a character or not within certain probability. Their second classifier performed a simple statistics calculation on scores to classify a group of regions as text/non-text group.

Gaddour et al [21] presented a modified MSER algorithm to detect Arabic script called Color homogeneity Region [22]. Their method uses two thresholds for each color channel instead of one used by MSER, to extract candidate region by minimum and maximum surface limit. Then, they calculated geometrical features associated with vertical projection histogram to filter Arabic text from non-Arabic ones. The

histogram that modeled the ligature characteristic of Arabic word is shown in Figure 1.6 with one experimental result of their system in Figure 1.7.

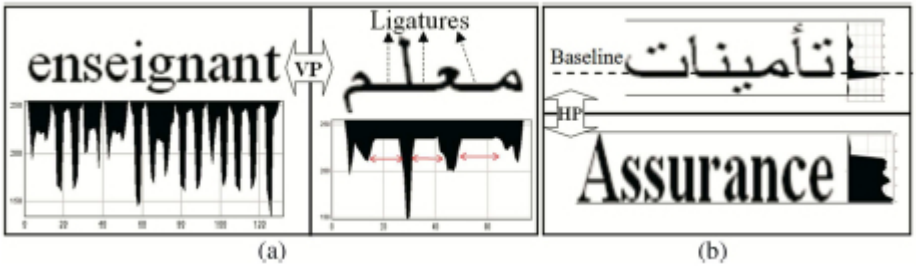

Figure 1.6: (a) Vertical projection; (b) Horizontal projection (HP).

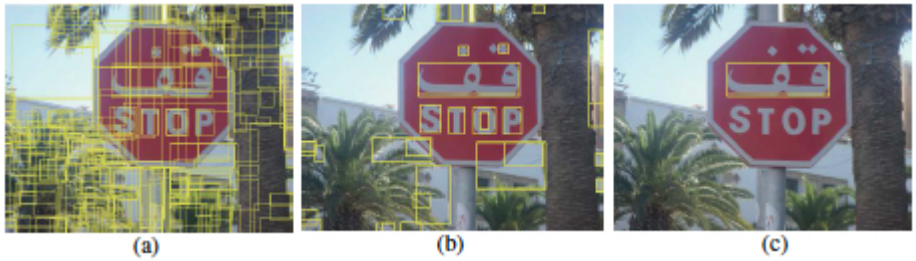

Figure 1.7: (a) All of the regions detected; (b) Region after filtering; (c) filtering according to Arabic features.

Using MSER [15] and SWT [6] algorithm we find the work of w. Ding et al [23] where they used MSER algorithm to generate candidate regions. For each region they calculated SWT to localize textual component where a hierarchy was constructed using seed growing approaches. Then, they used Convolutional Neural Network to filter the regions as characters or not with greedy iterative search as grouping characters search algorithm to form text lines.

1.3 Texture-based Method:

The texture-based method [1, 24, 25, 26] used in general the concept of sliding windows as search method to extract patches with multi-scale windows and then analyze it by exploiting the property of similarity and high intensity of text region using

Gabor filters [24] or wavelet transform coefficient [1] or Discrete Cosine Transform [25, 35, 36], where features are calculated for text detection or classification.

One of the interesting methods was of Chen et al [26] that proposed an automatic system for detection and recognition of text from natural scenes, their approach embeds a multiresolution and a multiscale edge detection to detect text in different sizes where at first stage they used edge-based features for text detection region by applying a multiscale Laplacian of Gaussian (LOG) edge detector that return a set of edge patches. For each patches they calculated a features like size, intensity, mean, and variance to eliminate some of them basing on certain criteria and then the remaining ones was passed to a recursive procedure that merged the adjoin edge patches with similar properties.

The high intensity contrast in gray scale images and the color difference in foregrounds and backgrounds of the text made it distinguishable, they exploit these proprieties for the second stage of their method where they applied a marginal distributions Gaussian mixture model (GMM) on a color space for foregrounds and backgrounds to describe each character separately rather than the entire sign for chines characters' text. Their choices to model each character aim to prevent the change in lighting that have impact on whole sign. They calculated the RGBHI distribution of colors spaces as parameters for GMM to easily segment the characters showed in Figure 1.8, also a confidence parameters was calculated to measure the segmentation performance of the component in each subspaces. They applied next a layout analysis to align characters using two cluster features (intrinsic and extrinsic). The intrinsic features were font style, color, and contrast that not change with the camera position. The extrinsic features were character size, sign shape, etc. that change with the camera position. This alignment helped to recover deformation of the sign if there exists.

In the third stage of their method they used an affine rectification to recover deformation of the text regions caused by an inappropriate camera view angle, they

used for that a non-texture based method where they calculated the normalized spatial direction from spatial parallels lines keeping on consideration the cases where the projection of lines in image plane are parallel or not. Then they reconstruct a front view of the sign from an affined text image by calculating the normal of the sign plane under the camera coordinate and using B-spline interpolation.

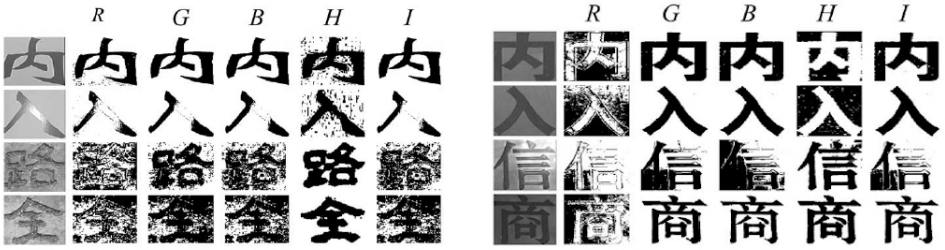

Figure 1.8: Segmentation using different components.

The approach of Slimane et al [27] consisted on printed Arabic text recognition using Hidden Markov Models (HMM) improved by a minimum and a maximum duration models [28]. The improvements contributed to build a system working in open vocabulary mode, without any limitations on the size of the vocabulary. Their system was composed from two stages, training and recognition based on Hidden Markov model Toolkit (HTK). They proceeded first by features extraction of words that were already extracted from images. The word vector features was calculated from sliding window passed along the word from right to the left shifted by one pixel, where for each window the number of black/white of connected component and the ratio between them, the perimeter, compactness and gravity center of the windows were calculated. To add more features to the word vector they normalized the size of windows to 20 pixels height, then they computed the horizontal and vertical projection values resulting in a vector of 53 coefficients.

They converted the extracted features vectors to a compatible file using HTK that initialized HMM for the learning stage where a sub-model was associated with each trained word. They choose the procedure Baum-Welch iterative estimation in HTK

for embedded training, that evolve two operations alternatively proceeded where one increase Gaussian number and other re-estimate Gaussian parameters. Finally, at the end of training stage a duration models was derived to alter the HMM topology. In the recognition stage many transition from sub-model to another were allowed to recognize any word in an open vocabulary fashion using a Viterbi procedure in HTK that look for the best state sequence. some miss classification occurred representing a major drawback of their method. The big advantage using duration model for topology of HMM was the capability to reduce drastically the recognition errors of the system.

From the older works that used texture-based we find the work of li et al [29] for text detection and tracking in video frames. The first stage of their work was a sliding windows search. Therefore, for each windows they calculated Haar features fed to Neural Network for text or non-text classification, the detected text was then tracked across video frames. Using the same notion of Neural Network Lienhart and Wernicke from [30] calculated and scaled different edge orientation in images and presented them as input to NN for classification. They used also a region bounding box growing algorithm where each rows and columns are checked to be text or not within the bounding boxes. More recent works [31, 32, 33] use Convolutional Neural Netwok (CNN) for text detection basing always on sliding windows. As example we find the work of Yousfi et al[4, 32] where they used three machine learning methods for text detection. The first one was the Convolutional Neural Netwok (CNN) where no preprocessing have been done due to its cursive characteristics by extracting features from patches images and then classify those patches in same time. The second one was using two multi-exit asymmetric boosting cascade on Multi-Block Local Binary Patterns (MBLBP) [34] and Haar features to classify blocks to text or non-text. Their methods proved high rate performance on large database of Arabic TV channel videos. The following Figure 1.9 show two examples of their experimental results.

Figure 1.9: Example of text localization using CNN-based method.

1.4 Hybrid based Method:

Hybrid method is the mixed of CC-based and texture methods that can merge between the advantage of the both and is more popular like CC-based than sliding windows.

Moradi and Mozaffari [25] proposed a new hybrid method for Farsi /Arabic text detection and localization in video frames. In the first stage of their method they extracted edge and stroke from I-fram video using Sobel edge detector that result in high-edge density in text areas. Then, they performed some statistics on the stroke in 4 directions (vertical, horizontal, diagonal and anti-diagonal) to eliminate a non-text edges, as result they obtained four text stroke maps for each directions. Next they exploited the characteristic of periodical gradient existence in Arabic/ Farsi text by creating a set of artificial corners from the intersection of the four dilated maps strokes then, they applied a Gaussian kernel in image to reduce the sensitivity of font where they had as output an image with higher artificial corner density. From this image a horizontal and averaged value histograms was calculated with the number of artificial corners to remove the histogram bins that are less than average representing the non-text region, finally the font size was estimated by calculating the average width, where if the estimation was 50% larger or smaller than the initial font size (

fixed to 20 pixels height in their approaches) they rescale the estimated fonts until they became with the same size of the proposed initial font.

The second stage was extracting the texture of the image following the intensity, they applied for that a Discrete Cosine Transform (DCT) on the gray-scale input image, the coefficients of DCT were determined for extracting Farsi/Arabic text lines and combined with a selected quadratic weight that gave the frequency importance of component. Next, they introduced an operator that measure the local contrast for efficient texture classification called Local Binary Pattern (LBP). They applied LBP on gray-scale image where another image was created in which each pixel value represented the texture pattern of the respective pixel. For more precision and for describing edge patterns they used a horizontally emphasized LBP (HELPB) and a vertically emphasized LBP (VELBP) operators to extract horizontal text strokes pattern and vertical text strokes pattern respectively the process is shown in Figure 1.10. The threshold of both HELBP and VELBP was adaptive, calculated using probability density function of the image's DCT coefficients [35,36].

The final stage consists on text identification using a SVM with radial basis function RBF. The SVM had as input a set of hybrid features extracted from both a texture intensity image and artificial corners image for each block of the height equal to the font size and width equal to font size *2. These hybrid features represented the energy, entropy, homogeneity, inertia, mean, variance, third-order central moment and the background complexity of each block. Finally, they performed a horizontal and vertical projection on the DCT coefficient-based texture intensity image to estimate the bounding box of text region.

In the work of Zhao et al [2], they presented a robust hybrid method that use both top-down and bottom-up processing for text detecting in natural images. first they used learning-based Partial Differential Equations (PDEs) [38, 39] to construct confidence map in faster way, where the complexity of this method is much better then sliding windows and is order only O(N), they applied on this map a connected component clustering to detect text region. Then, they used a two-level classification

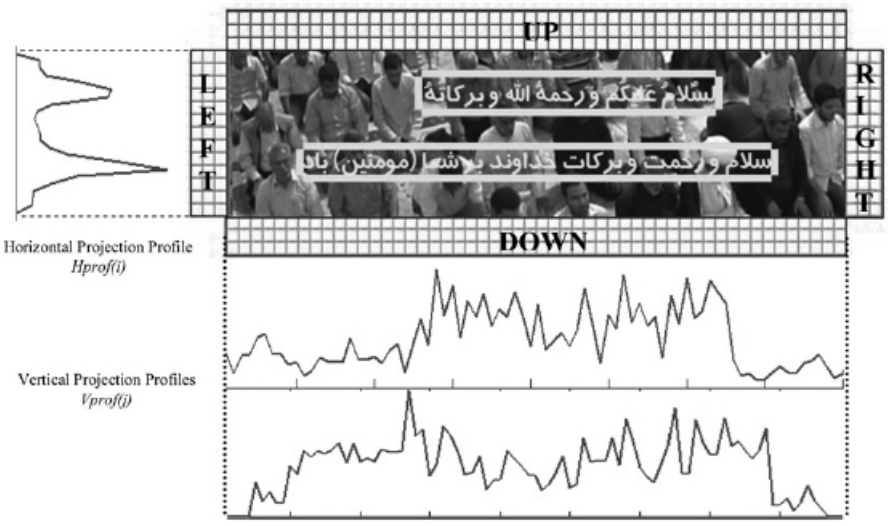

Figure 1.10: Candidate text region with corresponding horizontal and vertical texture projection.

to filter non-text regions.

Pan et al [40] they presented a hybrid approach to detect and localize texts in natural images. They first estimated the scale information in image pyramid and text position to detect text region, then they used a conditional random field (CRF) model with supervised learning to filter the non-text regions. At final stage they grouped regions filtered into text lines using energy minimization method. The following figures 1.11 and 1.12 shows the main processing of their method.

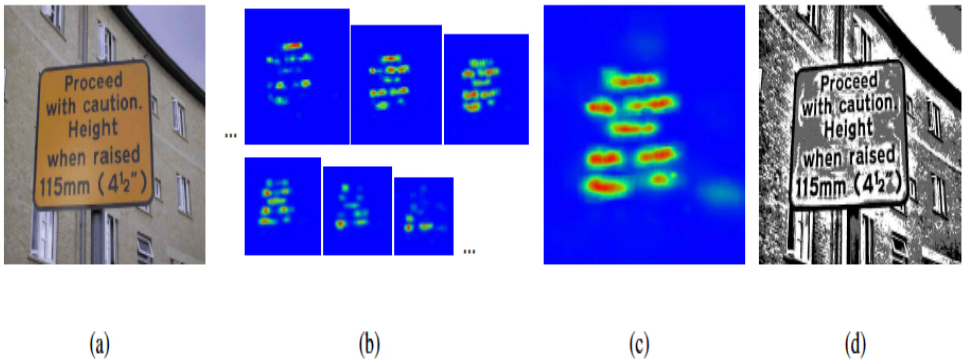

Figure 1.11: (a) Original image; (b) text confidence maps for the image pyramid; (c) the text confidence map for the original image; (d) binaries image.

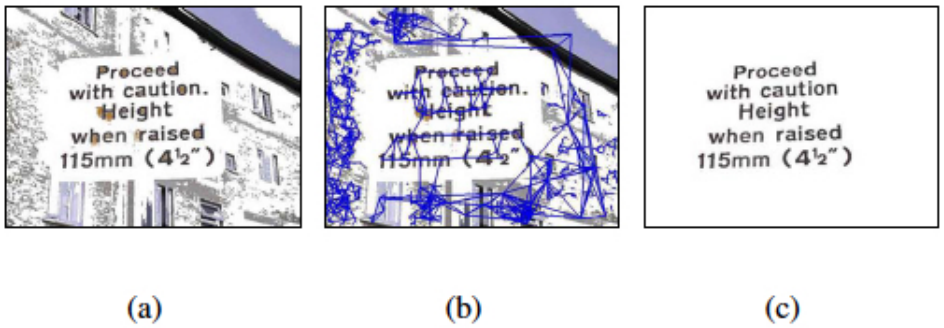

Figure 1.12: (a) component neighborhood graph; (b) components text with the learned CRF model.

1.5 Comparative studies:

The main advantage of Region-based techniques is time performance they can achieve a high precision in text detection much less then texture based techniques that consume more time for the search of text using high scaled sliding windows. Also, due from the low-level process they can segment mutiscript in different orientation and scale in contrast of texture-based that demand a training in specific languages to detect text.

For the advantages of texture-based methods we find a good localization of the text even in images with low resolution where we cannot differentiate background of the text from foreground specially in video sequence images. However, this advantage remain disadvantage for region-based method where we cannot easily segmentate and localize the text. Following the advantages of the both technique a Hybrid method was introduced from many researchers for a better performances and results.

In the following we present a comparison between the performance of methods that we described above in each three approaches.

For region-based we have three comparison following the nature of dataset used to test the performance of each methods Ding[23], Newman and Matas [12] and Yin et al [1] they evaluate their approach using a public data set of ICDAR11 [37] where Ding method showed more performance then Newman and Yin et al. the performance of each one are respectively 86.3, 86.29 and 73.1 as precision. 77 , 68.26 and 64.7 as recall. 81.4, 68.7 and 76.22 as f-score. For the ICDAR 2013 [41] the same performance of Ding was better then Yin. For the MSRA-TD500 [9] dataset, Yao et al [9] and Gomez [17] methods showed a very remarquable performance with large difference than Epshtein [6] and Chen. The performance of each method are given respectively 63 , (58, 54 Gomez), 25 and 05 as precision and recall with 60,56, 25 and 05 as f-score. For texture-based method the work of Yousfi [32] was tested in two kind of dataset called ES1 and ES2 without a comparative study with other methods. Their ES1 dataset was an ensemble of 201 images taken from Al-Arabiya, Al-Jazeera and France 24. They tested on ES1 their three approaches. The first approaches was detecting text using CNN, the second was using Haar-ada and the last one was MBLBP where CNN proved more efficiency then others where respectively the result are 75, 66 and 25 as precision. 77, 75 and 72 as recall. Finally, 76 ,70, 37 as f-score.

Slimane et al [27] evaluated the impact of duration models on HMM using synthetic database of word images composed of 20630 words. Their best results for recognition without using the duration model was 93.1% obtained with 6 states for

each sub-model. For minimum duration models they evaluated two cases, in the first case the value of minimum duration was measured manually from all letters and used in system where the rate of recognition was 81.5. In the second case the value was inferred from training stage and the recognition rate was 91.9%, the advantage in introducing such minimum duration was for less memory footprint and less CPU calculation.

Their experiment was on a varied database composed of news video from different Arabic channels. The recall and precision for TV7 Tunisia were 89.66% and 88%, for Al Jazeera 90.53% and 93.3% and for Al Arabia 10 hours 93.45% and 91.22% with all duration about 10 hours. For the hybrid-based method of Moradi and Mozaffari [25] trained their SVM using 4917 Farsi/Arabic text lines, then they evaluated their approach on their own dataset of 50 videos containing different clips, movies, and animations where recall, precision and f-score were respectively as follow 91.38, 87.22 and 89.25. The Yin et al [1] region-based method with scheme-IV showed more efficacy than Pan et al [40] hybrid-based method for the Multilingual dataset where respectively the results are for the precision's 82.63 and 64.5. Then, for the recalls are 68.45 and 65.9. Finally, for the F-score are 74.58 and 65.2.

Chapter 2

ARABIC DATA BASE

In this chapter we present an overview of different Arabic databases from character OCR to real world images passing by handwriting and at the end we describe our own database with its ground truth and evaluation protocol.

2.1 Introduction:

Arabic is a language that is spoken by more than 422 million people around the world [42]. Due of its rich vocabulary and its importance for all Muslim in the worlds, developing computer and mobile application to detect and recognize Arabic script remain indispensable nowadays. In general, text script database is the first stone for the development of any such identification and recognition systems. However, for Arabic language we find few real world database unlike other languages in the world where we find a benchmarks specially for Latin languages like ICDAR [37, 48], MSRA-TD500[9], Chars74K [43], IIIT-5K Word [44] and SVT [45]...etc. The optical character's recognition OCR and handwriting Arabic databases had more attention then real world scene Arabic text images, these databases were used for bank check processing, mail sorting, signature verification...etc. More details about the nature and the size of theses database are described in details in section 2.3 of the chapter with the few real world scene video text images databases like Alif [46], AcTiV [47] AcTiV 2.0 [58] that are used more for recognition with free segmentation due to the

nature of text in video frames that we can't diffirenciate it eassly from backgroud. Last year Jain**[31]** and Tounsi**[49]** presented their real world scene text images that can be used for text detection. However**,** For the lack of this kind of database, we also developped our own constitued from 220 images with their ground truth and we defined a performance protocol that mesure precision, recall and accuracy.

2.2 Arabic language characteristics:

The nature of Arabic language is a cursive language where characters are connected constituing a word or a sub-words in the language vocabulary, written from the right to the left. This connection is designed in manner forming an imaginary line, called baseline as Figure 2.1. The Arabic language vocabulary is very rich and complex where a single word that have a specific meaning can be composed by a multiple sub-words with other meaning therefore a recognition process remain complex. These sub-words are formed from a set of 28 main characters, each character can have different shapes following their position in the word (begenning, middle, end) where seven of them cannot be joined to their left neighbours. We find three letters have three dots (ش ,ث) four letters have two dots (ق ,ت ,ي ,ة) and eight letters have one dot (ظ,ز ,ن ,ب ,خ ,غ ,ف ,ض). The position of these dots are very determinant to the meaning of the words, a simple changing in position of one dot give totally a different meaning for characters where region of character stroke remain the same but the meaning and pronunciation are totally different "Ba: ب " and "thaa: ت ".

Also the word can contain "Hamza: ء" called secondary characters (complementary), like these four characters, which can take the secondary nature " alif : أ, إ", "waw: ؤ", "kaf: ك" and "ya: ئ".Finally, they are a specific diacritics marks for Arabic language that we find in the top or bellow letter and represent a short vowels like Fat-hah, Dhammah, Kasrah , sukkun, Tanwen and shaddah.. Fig 2.1 lists these diacritics. Also in Arabic language there is not uppercase and lowercase but there are a long letters and short letters in general the long letters are composed from two

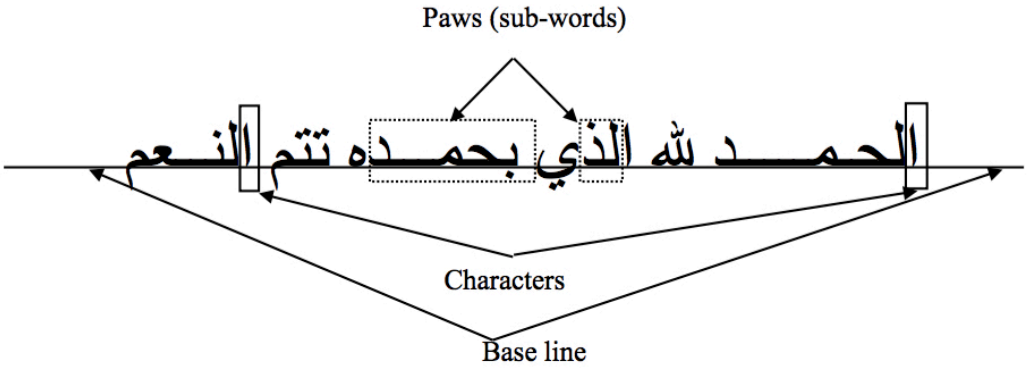

Figure 2.1: Arabic words component

letters to prolonge the sound of the initial letter like “Laa :لا ”, “mii: مي” , “boo: بو” . The shape of letter in Arabic words can change following its position at the beginning, middle or at the end as figure 2.2 shows.

2.3 Arabic Databases:

Due to the intracacie of arabic language the process of its detection and recognition in natural images is hard. For those process a database containing arabic words from language vocabulary must be contructed. Several work was introduced these recent years by many researchers specially for hand text recognition and photo OCR. Where we find the work of Muhtased et al **[42]**presented for preparing databases and benchmarks for Arabic text recognition research a novel minimal Arabic script that covers different shapes of Arabic alphabet in all positions. The importance of their script was the ability to cover a total of 125 different shapes in only three lines ensuring a minimum occurrence for each letter. The script can help to build a more accurate recognizer trained with dataset that contain hand writing text with their minimal script collected from different writers. Due to the lack of public Arabic Text database their script can help in constructing

No	Name	Isolate	Beginning	Middle	End
1	Alif	ا	-	-	ـا
2	Baa	ب	بـ	ـبـ	ـب
3	Taa	ت	تـ	ـتـ	ـت
4	Thaa	ث	ثـ	ـثـ	ـث
5	Jeem	ج	جـ	ـجـ	ـج
6	Haa	ح	حـ	ـحـ	ـح
7	Khaa	خ	خـ	ـخـ	ـخ
8	Daal	د	-	-	ـد
9	Dhal	ذ	-	-	ـذ
10	Raa	ر	-	-	ـر
11	Zaa	ز	-	-	ـز
12	Seen	س	سـ	ـسـ	ـس
13	Sheen	ش	شـ	ـشـ	ـش
14	Saad	ص	صـ	ـصـ	ـص
15	Dhad	ض	ضـ	ـضـ	ـض
16	Tta	ط	طـ	ـطـ	ـط
17	Dha	ظ	ظـ	ـظـ	ـظ
18	Ain	ع	عـ	ـعـ	ـع
19	Ghain	غ	غـ	ـغـ	ـغ
20	Faa	ف	فـ	ـفـ	ـف
21	Qaf	ق	قـ	ـقـ	ـق
22	Kaaf	ك	كـ	ـكـ	ـك
23	Laam	ل	لـ	ـلـ	ـل
24	Meem	م	مـ	ـمـ	ـم
25	Noon	ن	نـ	ـنـ	ـن
26	Haa	ه	هـ	ـهـ	ـه
27	Waaw	و	-	-	ـو
28	Yaa	ي	يـ	ـيـ	ـي

Table 2.1: Arabic character forms

both hand writing and typed databases using different fonts and size.

The motivation that was behind constructing a minimal Arabic script was their analyzing of some available database, where they discover that some character appeared 50 times more than other characters. The difference in occurrences have a great impact in rate of recognition. They gave a concrete example to prove that lack, by referring to a trained Hidden Markov Model recognizer using an arbitrary 2500 lines of Arabic text where some character had low occurrences around 8 and other had over occurrence around 1000 occurrences.

In their paper they cited the characteristics of Arabic language as cursive language with 28 basic alphabets where some characters have four different shapes following their position in the word (standalone, initial terminal, medial form). The diacritics (short vowels) are often present in Arabic words either on top or below of the letters. A simple changing of the position of this diacritics may change a complete meaning of a word. The Standardization and Metrology Organization ASMO-449 and ASMO-708 with ISO 8859-6 defined 36 Arabic letters for easy use in computer and following the nature of Arabic letter that can be written in four different shapes, the set of alphabet can be extended about 126 characters to represent all basic alphabets with their different shapes too for most of them.

The CORPORA database defined by Muhtased et al contained 4.25 million characters, used to analyze and extract the minimum script. Their database was created from Arabic text of Arabic lexicons of two Hadith books and another lexicon containing the meaning of Quran. The algorithm that was deployed to extract the minimum script, proceed by a random selection of the word in CORPORA database, then by a validation of the word using a simple test, if the word not belong to the set of minimal text, then a contextual analysis algorithm was used to decode the word in a proper letter shapes, after they checked if there were multiple occurrences of letters in the selected word, if the case the word was rejected. Finally, their algorithm checked each letter of the word with the letters table that contain different shapes

of Arabic alphabet, if one entry of table was flagged then the word was ignored otherwise the word was added to the minimal text and the process was repeated until all letters in table flagged. Their algorithm was modified several time taking in a count several criteria for more improvement of minimal text. Their final minimum script is not unique it can be constructed from different others words.

Another work of Slimane et al [50] where, they proposed a new database called APTI contained Arabic printed text images in large scale around 45313600 single words (250 million characters) synthetically generated using a lexicon of 113284 words, 10 Arabic fonts, 10 font sizes and 4 font styles. The text font selected for ATPI database contained different complexity for representing Arabic word, started from simple font with less ligature and overlap to complex one with multiple ligature and overlap. The 10 used fonts were Andalus, Arabic Transparent, AdvertisingBold, Diwani Letter, DecoType Thuluth, Simplified Arabic, Tahoma, Traditional Arabic, DecoType Naskh and M Unicode Sara. These fonts were represented in different size (6, 7, 8, 9, 10, 12, 14, 16, 18, 24 points) merged with different style (plain, italic, bold, italic and bold). The text images were automatically generated from a high resolution gray-scale input images down-sampled to a low resolution using antialiasing filtering, with different variability in fonts, sizes, styles, height of each word image also with various forms of ligatures and overlaps...etc.

They associated an XML file containing a ground truth information about characters in detailed way with each image word, where they specified the transcription of the word, the number of Paws and its sub-elements with the sequence of characters, also they specified the name font, style and size. The XML file contain also some fields left for future extension of database, like field containing effects (noise, deformation...) that may be added in future, plus the name of procedure that generated the specific word image, for their ATPI database they used only one procedure called in XML file downsampling5. The database was divided into six equilibrated sets; each set contain a near number of occurrence of each letter within different words. The five first sets are available for the scientific community and the sixth one set

was kept internally. Their six sets division was generated automatically using a distributed procedure that counted the number of letter occurrences in each word and then classified words in each set. This distribution remains very important to improve the accuracy of any classifier that use their database, because that allow to know which character is under-represented and try to recover the lack. They associated with their database a set of robust benchmarking protocols (20 protocols) to evaluate its use, these protocols proposed a well decomposed sets of training and testing dataset, each set have a specific and objective aim, for example one of their protocol aim to measure the capability of system to recognize muti-font text. . . .etc. .

We found also some others works [51, 52] where in theirs papers they gave an interesting statistic about some Arabic text databases. They mentioned the source, quantity, nature and usage of some of them. Most of these database unfortunately are not for public access and was prepared for specific domain like bank-checks, signature verification...etc. The following are some cited examples by them:

1. A database containing 26459 Arabic names of 937 Tunisian town/village, handwritten by 411 different writers. [53, 54]

2. A public limited database constructed by 100 writers, each one contributes by writing 67 literal numbers, 29 most popular words, three sentences representing numbers/quantities used in checks, around 4700 handwritten words produced by 100 writers [55].

3. A small database for digits where 17 writers contributed by writing 10 digits 10 times [56].

4. A 220000 handwritten forms filled by more than 50000 writers was used to construct Arabic and Persian isolated characters database [57].

5. Database contain 29498 images of sub-words, 15175 images of Indian digits and image samples from 3000 real-life bank checks [59].

6. A database for bank-checks with 70 words of Arabic literal amounts extracted by 100 writers from 5000 checks [60].

7. Printed database consisting of 946 Tunisian town names [61].

8. A database containing 360 handwritten addresses of 4000 words [62].

9. A database consisting of more than 17820 names of 198 cities of Iran [63].

10. A database with signatures containing 37000 words, 10000 digits, 2500 signatures and 500 free-form Arabic sentences [64].

11. DARPA Arabic Corpus consists of 345 scanned pages of printed text in 4 different fonts [65].
    For the databases that contain natural images and scene scripts video images with their associated annotation to calculate their performances and comparison we find the following examples.

12. AlIF dataset include 6 532 text lines images cropped from video frames of Al-Arabiya, Al-Jazerra and Tunisia news channels [46].

13. AcTiV dataset contain 21 520 text lines images from video frames of Arabic news images [47].

14. Google Images offer freely a 2000 Arabic words images with natural background.

15. ARASTI database constructed from 374 images with 1687 words cropped and 11615 characters segmented [49].

2.4 Our Database:

For the development of our system we were obliged to construct our own database, due to the lake of real world images database that contain Arabic scripts. However, the process of detection of Arabic script in natural images is very challenging due to the nature of the languages and its large vocabulary too. Therefore, for the nature

of the script the detection process can omit small important characteristic of Arabic characters like diacritics and points. Also, for the large vocabulary between 60, 000 and 12, 000,000 different words [66] compared with other languages can lead to serious problem in size to construct databases for both detection and recognition processes. Hopefully, for scene text world images we are subject to a small set of words that can be repetitive. For that we collected our database from natural images that contain Arabic text and few English text too, taken from different sources like indoor, outdoor, books, brochures...etc. These images are photographed under different conditions. The following Figure 2.3 show different images that contain Arabic script

Figure 2.2: Images taken from different sources

For each image in database we associated a ground truth image that have been generated semi-manual as shown in Table 2.2 below. These images are trivial for calculating the performance of our system and others too, using a specific protocol associated with it.

Original image	Ground truth
	
	

Table 2.2: Original image with its ground truth

Therefore, we created two training dataset for our each two kind classifiers. The first training dataset was constituted from positive and negative examples that represent characters and non characters cropped from images. The second training dataset represent words and region text cropped too from images. Further, to expand our training dataset we used synthetics words of Slimane [50] dataset. For each words with each extracted characters, words and text lines we applied some image processing like rotation with different angles, changing background, dilatation, skew, scaling etc.

The training dataset that feed character and non- characters' classifiers was constituted from geometrical descriptors calculated on 4324 images cropped and 11 713 rendered total of 16037 images. The same for the training dataset that feed text and non-text classifiers a set of statistical descriptors was calculated on 77370 images cropped and rendered with image processing functions. The Table 2.3 and Figures bellow gives an overview for some characters and words with theirs main geometrical and statistical values

Characters	stroke_mean	stroke_std	std/mean	area	Perimeter	N_holes
	2.46139	1.219792	0.49557	510	43	1
	3.385405	2.069415	0.611275	651	52	0
	3.291951	1.80477	0.548237	644	65	0
	8.275565	5.286153	0.638766	490	37	0
	2.147783	1.088603	0.50685	313	62	1
	4.890975	2.784823	0.56938	2462	88	1
	4.754728	2.635169	0.554221	1494	108	0
	2.891414	1.534396	0.530673	316	38	0
Non-characters	stroke_mean	stroke_std	stroke_std /	area	Perimeter	Num holes
	1.675532	0.890874	stroke_mean	136	80	1
	1.292929	0.455106	0.531696	67	42	2
	1.449438	0.636197	0.351996	59	48	0
	1.49763	0.633761	0.438927	156	77	3
	1.447155	0.756578	0.423176	104	40	2
	3.608794	2.464918	0.522804	925	74	2
	1.28125	0.51444	0.683031	64	43	1
	1.461929	0.6943	0.474921	140	81	2

Table 2.3: Geometrical descriptors for positives and negatives examples of our first training dataset

Figure 2.3: Words images examples extracted from images in database

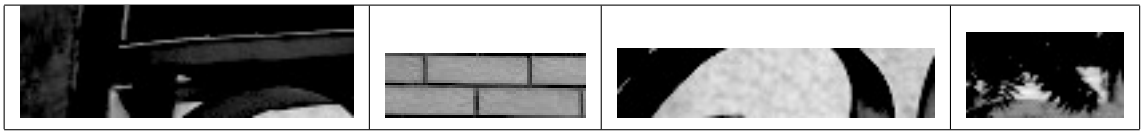

Figure 2.4: Non text examples

Figure 2.5: Synthetics words in the left transformed using rotation in middle and changing background in the right

2.5 Evaluation protocol:

In some real world problems, calculating the performance of system basing only on accuracy is not a trivial metric. Like our problem where the performance was most based on the precision and the recall to a real evaluation of our system. Therefore, that leaded us to construct our own ground truth images associated to each one in database. Although, any detection system cannot exactly extract all text regions like ground-truth ones, for that to measure the performance we base on precision because only positive regions (character/text line) interest us for the process of detection,

Also the recall that tell us how well our system do to find positive regions.

The performances of our system was calculated basing on pixels' information. Therefore, the detection process performance description using low level pixels can result in four cases as confusion matrix shown in Table 2.4 where the true positives pixels are the ones correctly predicted as region (character/text) comparing with the ones in the ground truth [67]. The false negatives are the ones predicted as non-region (character/text) and they are really the pixels that belong to region (character/text) in the ground truth. For the true negative they are pixels that not belong to region(character/text) and predicted correctly in contrast of False positive where they are predicted as region(character/text) and they are not comparing always with ground truth.

		Predicted classes	Predicted classes
		y=+1	y=-1
Actual classes	y=+1	True Positive	False Negative
	y=-1	False Positive	True Negative

Table 2.4: Confusion matrix.

In general, the precision in our case was defined as the ratio between the sum of true positive predicted pixels as text and the total number of pixels predicted as region text [68, 92].

$$precision = \frac{true\,positives}{true\,positives + false\,positives} \tag{2.1}$$

For the Recall was defined as the ratio between the sum of true positive pixels and total number of positive pixels belonging to ground truth images.

$$recall = \frac{true\,positives}{true\,positives + false\,negatives} \tag{2.2}$$

The Accuracy was calculated as ratio between true positive pixels defining detected text plus true negative pixels defining the background over the sum of all pixels in image.

$$accuracy = \frac{true\,positives + true\,negatives}{true\,positives + false\,positives + true\,negatives + false\,negatives} \tag{2.3}$$

Finally, the good classifier is the one that can balance between a recall and precision. The excellent classifier is the one that can lead both precision and recall to the value 1.0, well this is not reached in the real world problems like our problem. We find during our test of classifiers that we can have two extremes that we get rid of them. The first extreme called in optimistic classifier where we had very high recall and low precision due that classifier predict all kind of positive pixels (false ones and true ones). The second extreme called too pessimistic classifier was in the case where fewer positive are predicted but with the high confidence that are only true ones, such case leaded us to high precision but low recall.

Chapter 3

THEORY

In this chapter we present The theory used to develop our systems starting by segmentation of region as preprocess step using the Gomez notion. Then we will explore machine learning methods by defining the concept of Adaboost then Neural Network and finally Support Vector Machine.

3.1 Introduction:

Today, data amount presented is huge with increasing of technologies. Therefore, to deal with this data for better understanding of human world, the classical methods are not suitable. for that a new method are deployed called machine learning, that have capability to deal and learn a considered amount of data and present a suitable result. One recent definition of machine learning is introduced by Tom Mitchell, where he says: "a computer program is said to learn from experience E, with respect to some task T, and some performance measure P, if its performance on T as measured by P improves with experience E" [69]. In the field of image processing specially scene text detection many method of machine learnings were used, specially in the case of texture based detection, when they used sliding windows to extract patches fed to machine learning methods for a specifics tasks following researchers interest. In our systems we used region-based detection notion to first preprocess our images and get meaningfulness regions probable to be region text. Then, we apply some calculation

on these regions to feed machine learning methods (SVM, NN, AdaBoost), where these latter are trained on labeled data to detect if region is character/text. The labeled data used by supervised learning algorithms (machine learning concept) are the ones used by our systems for classification. Where we defined two sets of training sets and each example of training sets are associated with the label defining its as character/Non-character or text/non-text. In the following we will give definition of essentials theory that we used.

3.2 Gestalt theory:

Gestalt theory inspired from cognitive science start with the fact that human retina identifies a subset of grouped objects. These objects are grouped by active laws in visual human perception. The "enigmas of perception", introduced by Gaetano [70] consists on identification of certain subgroup of perceptum and some physical object following laws and principals that describe Gestalt theory. This theory is based in general on two main organizing laws the first one is grouping laws where agroups are constructed from image, starting from atomic level described by characteristics like color, shape or direction. For the second one is the conflicts laws, where the conflicts can occur in grouping laws and lead to different interpretation, specially if the grouping laws compete where one win and omit the other, or they act simultaneously giving an overlapping groups or noting. For grouping laws are described by some principal in computer vision like Shannon sampling principle, Wertheimer contrast invariance principle and Helmholtz's principle. Gaetano [70] presented a basic list of grouping laws that can be applied in a recursive way from atomic data to global gestalts passing by partial gestalts. This list is presented as follow:

Vicinity: mean connectedness of spots or cluster when distance between them is very small comparing to the Rest (Figure 3.1).

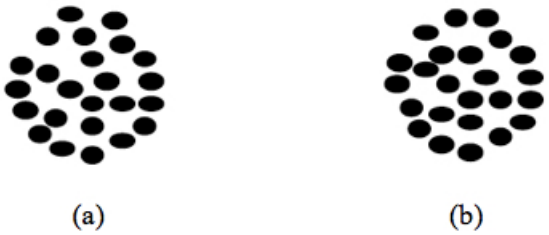

Figure 3.1: Vicinity illustration in (b) by adding point to (a)

Similarity: group similar objects from low level to high one, following color, shape, texture, orientation (Figure 3.2).

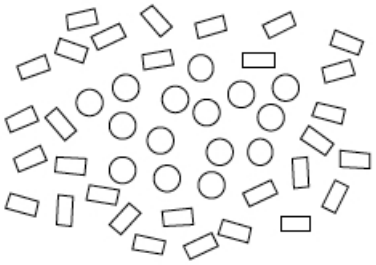

Figure 3.2: The similarity is illustrated by two homogeneous regions (circles, rectangles).

Amodal completion law: applies when T-junction is detected in general by the presence of some object occlusion in images. Where, the leg of the T is then extrapolated and connected to another leg in Figure 3.3 show good continuation law (same direction) by the connection of two T-legs.

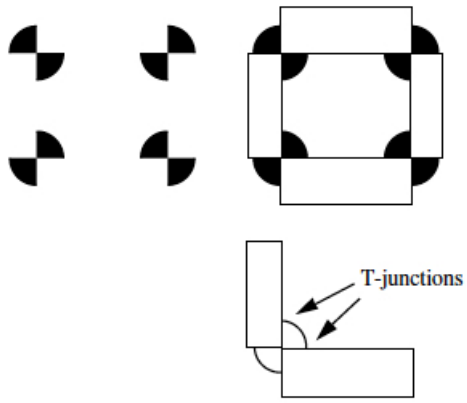

Figure 3.3: T-junctions legs connected constituting an amodal completion.

Closure: based on constitution laws, which an object is defined by an interior part of plane surrounded by a closed contour. Where the exterior part defines a background. As shown in Figure 3.4.

Figure 3.4: The interior of the curve is an object and its exterior is the background.

Constant width: this laws is often used to group two parallel curves, that define the boundaries of a constant width object. Figure 3.5 illustrate this laws.

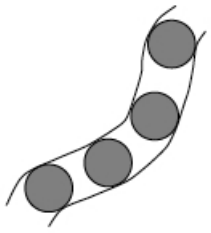

Figure 3.5: Width constancy law applies to group two parallel curves.

Tendency to convexity: The convex contours define the boundary of a convex body even is not connected. Figure 3.6 illustrate convex curve defining circle on the black background.

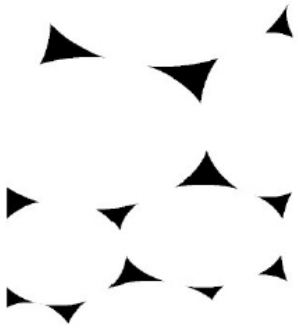

Figure 3.6: White circle curves on black background.

Symmetry: used to group symmetric objects to a straight line as Figure 3.7 show.

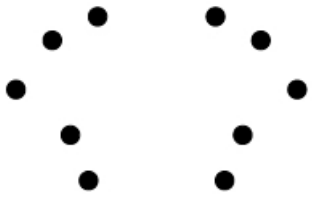

Figure 3.7: Symmetry law illustration.

Past experience: we can group object that define for example circles, rectangles...etc. in such familiar way that they were in our past experience.

However, these grouping laws consist on geometrical statistical calculations that belong to primary process by the visual system. But nowadays, no one presented a biological explanation on why they work. In computer vision several works addressed this calculation to solve many problems [71,72,73,74] In the following we will present one principle that define Gestalt theory and used by Gomez to construct meaningful regions.

3.2.1 The Helmholtz principle and hierarchical clustering:

The Helmholtz principle can be defined in two contrarious sense The first one is "we do not perceive any structure in a uniform random image" [75]. The second one is "whenever some large deviation from randomness occurs, a structure is perceived". Attneave [76] was the first gestaltist that used the first sense in his research about the random noise digital image. The second definition are used also by many researchers like Dosolneux [72] where Gomez and Karataz derived their approaches to extract meaningful regions.

Helmholtz presented three examples to show the perception principal in his research study. Therefore, to show the concept of visual grouping following various modalities like same color, orientation or position...etc. Helmholtz assumed that atomic objects O1 , O2 , . . . , On are present in an image with k of them, O1 ,. . . , Ok that have a common feature. This assumption lead to a dilemma: "Is this common feature happening by chance or is it significant and enough to group O1 , . . . , Ok ?". He used a mental experiment to answer this question where he assumed a priori that the considered quality had been randomly and uniformly distributed on all objects O1 , . . . , On . then mentally assume that the observed position of objects in the image is a random realization of this uniform process. Where he finally ask the question: Is the observed repartition probable or not? If not, this proves a

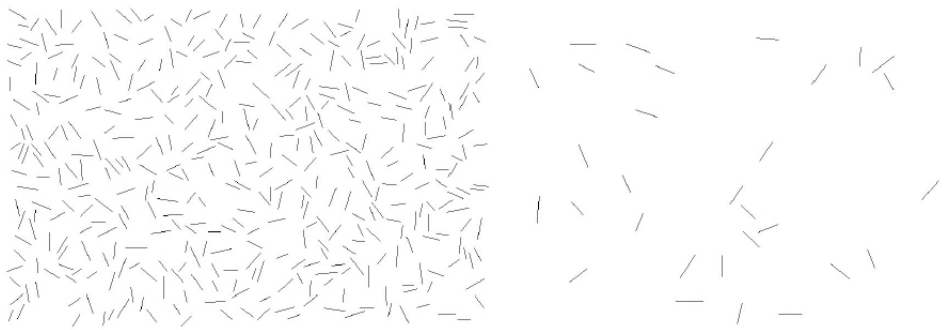

Figure 3.8: The Helmholtz principle in human perception: Left contains no meaningful alignment, while the right one contain meaningful alignment

contrarious that a grouping process (a gestalt) is at play. The Helmholtz principle states that the grouping of these object are significant if expectation of the observed configuration O1,..., Ok is very small in image (Figure 3.8). He used for that the following Definition:

Definition : (*ε-meaningful event*). *We say that an event that is ε-meaningful if the expectation of the number of occurrences of this event is less than under the a-contrario random assumption. When 1, we simply say that the event is meaningful.*

Figure 3.8, illustrate the true principle of Helmholtz where events is perceived if and only if they are meaningful (or Gestalt) respecting the above definition . To assess the ε-meaningful definition using mathematical model, he proposed to use the tail of the binomial distribution, assuming that a given object Oi has a considered quality equal to p. Under the independence assumption, the probability that at least k objects out of the observed n have this quality in common is random assumption. When $\varepsilon \leq 1$, we simply say that the event is meaningful.

$$B(n,k,p) = \sum_{i=k}^{n} \binom{n}{i} p^i (1-p)^{n-i} \tag{3.1}$$

Also, Helmholtz associated statistical testing noted by Nconf (number of different possible configuration) to the search of gestalts, where he considered that the event can be defined as ε-meaningful if the number of false alarms NFA(expectation of

number of geometric events happening by pure chance) is very small. The equation of NFA is given as follow:

$$NFA = N_{conf} B(k, n, p) < \varepsilon \tag{3.2}$$

Gomez [77] introduced this idea to construct a Hierarchical Clustering, where in each merge of its node he measured the meaningfulness by using the above NFA. In the following we will explore the Gomez work in constructing Hierarchical Clustering using agglomerative approach.

3.2.2 Hierarchical clustering:

Hierarchical clustering is method that group similar objects together in progressive way using Bottom-up or Top-down approaches. The first approach called also agglomerative tend to group similar objects from cluster containing single object to larger clusters using similarity function measure. This latter is represented in three popular methods, the first one group two clusters following the closest object (Single-link), the second one group two clusters using the farthest objects(Complete-link) and in the last one the groupement is based on average of all objects in cluster (Average-link). Therefore, for the Top-down approaches called divisive the hierarchical clustering is formed by partitioning the data into smaller clusters basing in general on k-means concept. Gomez in his research used agglomerative approaches to construct meaningful region in image. He chooses single-linkage algorithm to merge two clusters basing on gestalt law of proximity and similarity and Euclidean distance metric. However, we can use other two linkage clusters with others distance metrics calculation where that will lead us to definitively different results.

Gomez introduced the concept of maximal meaningful clusters using NFA to measure the meaningfulness in each branch of the tree. This concept get rid of the main weakness of Hierarchical Clustering that constraint us to use some heuristics to decide in the final results, These heuristics oblige us to select fixed number of clusters within the partition that have maximum lifetime in dendrogram.

Gomez used Evidence Accumulation to improve the results given by unsupervised

solutions, this method was introduced by Fred et al. [78]. He deployed this method to solve the problem of the final decision of meaningful clusters. Where the characters were grouped for text detection under different features(color, stroke width, size....etc.) resulting in different dendrograms that lead to the question about the best solution of the best feature clustering. Therefore we cannot decide on the best solution for one feature or a set due a to natural problems that can exist in images like (blur, distortion...etc).

Gomez used co-occurrence matrix (similarity matrix) D to vote on combined solutions of different clustering's. The voting formulas is given as following:

$$D(i,j) = \frac{m_{ij}}{N} \tag{3.3}$$

Given a set of N initial clustering's, mij is the number of times the feature vectors i and j are assigned to the same cluster among the N initial clustering [77]. Finally, a Hierarchical Clustering was applied on the co-occurrence matrix D to decide on final clusters.

Figure 3.9 show examples of Gomez experiment using maximal meaningful cluster on decomposition of maximally stable extremel Region (Seen in Chapter 1) MSER illustrated in (b). Two dendrograms were constructed following two features the first one is intensity mean of the inner region + x,y coordinates of the region center illustrated in (c) and the second one is stroke width of the region + x,y coordinates of the region center (d)

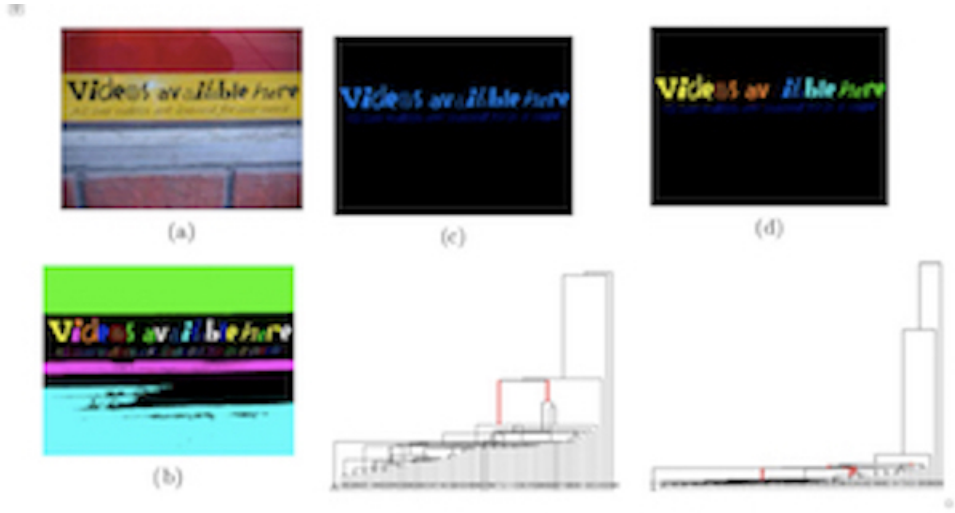

Figure 3.9: (a) Original image; (b) extracted MSER region; (c) and (d) text region with their associated dendrogram below.

3.3 Machine learning:

3.3.1 Adaptive boosting:

Adaptive Boosting (Adaboost) is an approach of Machine learning, introduced by Scharpire and Freund in 1996 [79]. Addressing the conjecture question of Kearns and Valiant in 1988 who was the first in this domain by asking question "Can a set of weak learners be combined to create a stronger learner?". Scharpire and Freund combined a weak learners to create a strong one using voting scheme to improve accuracy. AdaBoost contoured a great success with many applications, in biology, image processing, and speech processing. Adaboost learning algorithm is based on supervised learning called base or weak learning algorithm. The boosting process call this latter algorithm repeatedly to improve its performance. A labeled training dataset is presented to the weak learning algorithm producing a weak classifiers where their error rates are better than classifier based random guess in every prediction process. Therefore, this lead to assumption that the base learner produces a weak hypothesis representing the central study of boosting and called weak learning assumption. The

first Adaboost was based on majority voting upon three weak learners only and by the work of Schapire in [80], the elementary Adaboost was extended to multiple hypothesis [81]. The word 'adaptive' means that each weak learner (hypothesis) tries to overcome the mistakes of the previous one. This process is done by training each subsequent hypothesis on a new dataset in which the examples misclassified by the previous hypothesis.

The most used Adaboost algorithm is the one based on confidence-rated predictions, written by [82]. Where, he considered that AdaBoost is an ensemble learning technique, combined with weighted values result on a strong classifier H.

The Adaboost is trained given a set of N training examples where each input $x_i \in X$ is associated with $y_i \in Y$ as label and a number M of iterations (round). We deal here with classification problem for two classes -1 and +1 following our text detection systems.

We denote a weak learner as $h(x)$, so

$$\hat{y} = h(x) \tag{3.4}$$

Where, a set of weak classifiers are noted by $h_i(x)$ and can take the values +1 or -1. Then, the output of weak classifiers are combined using learned coefficient w_t. These latter are called confidence coefficient, associated to each weak learner given in equation 3.1, if the value of w_t is higher it mean that weak learner contribut most in final decision. w_i is base on weighted error, the less error is the higher is the confidence.

$$w_t = \frac{1}{2} log\left(\frac{1-\varepsilon_i}{\varepsilon_i}\right) \tag{3.5}$$

The equal weight assigned in first iteration to all training examples is denoted by$D_i^1 = \frac{1}{n}$. The training process is based on training each weak learner on a new dataset in which the weights of misclassified examples by the previous weak learner

are increased and the weights of the correctly classified examples are decreased by updating the weight of each hypothesis or weak learner. This training process by reweighting the weights show stability and efficiency [83].

Therefore, in each round t, the weak learner learn h_t to minimize the weighted misclassification error and using the voting weight this error is added to others.

$$\varepsilon_t = \sum_{i:h_t(x_i)\neq y_i} D_i^t \tag{3.6}$$

Then, also in each time step t+1, the distribution D^t is updated and normalized over all other distribution. The two equations are given as following:

$$D_i^{t+1} = e^{-y_i w_t h_t(x_i)} \times D_i^t \tag{3.7}$$

$$D_i^{t+1} = \frac{D_i^{t+1}}{\sum_{j=1}^{N} D_j^{t+1}} \tag{3.8}$$

In general, the learning process terminate when the maximum iteration M is reached or a weak learner with $\varepsilon_t < \frac{1}{2}$ cannot be found. The final prediction predicting on one example x_k of training dataset can be denoted as $H(x_k)$. And given by the following equation:

$$H(x_k) = sign(w_1 h_1(x_k) + w_2 h_2(x_k) + ... + w_i h_i(x_k) \tag{3.9}$$

So, the output prediction for all dataset X set of weak learners is given by:

$$H(x) = sign(\sum_{t=1}^{M} w_t h_t(x_i)) \tag{3.10}$$

Finally, the details of AdaBoost algorithm can be described as following :
Input: Training Data $\{(x_1,y_1\ ),\ldots,(x_N,\ y_N)\}$, Maximum number of rounds M.

Training:

$D = \frac{1}{n}$, for i=1,2,...,N.

for t=1 to M do

$\varepsilon_t = \sum_{i:h_t(x_i)\neq y_i} D_i^t$

Define a hypothesis $h_t$that minimizes ε_t and satisfies the condition $\varepsilon_t < \frac{1}{2}$

$w_t = \frac{1}{2} log\left(\frac{1-\varepsilon_i}{\varepsilon_i}\right)$

$D_i^{t+1} = e^{-y_i w_t h_t(x_i)} \times D_i^t$

$D_i^{t+1} = \frac{D_i^{t+1}}{\sum_{j=1}^{N} D_j^{t+1}}$

End for Prediction $H(x) = sign(\sum_{t=1}^{M} w_t h_t(x_i))$

3.3.2 Neural Network:

Artificial Neural Network was inspired from cognitive science, that studied human brain components and functions. The human brain is a complex and nonlinear system with highly parallelized processing information done by huge amount of connected components called neurons [82]. The biological structure of neurons are constituted from 1) dendrites where their structure looks like hair, they receives the signals from other neurons these signals are electrical impulse. 2) The cell body that contains the nucleus of the neuron, it makes a summation of the signals received as input and following the result obtained provides a current as output. 3) The axon, serves as conductors of electrical signals from the output of one neuron to the input (synapse) of another neuron these synapses are the connection points between neurons. The classical function presented by the biologist of the neuron cell is that a summation is done using the impulses nerve transmitted by the dendrites, if the result of this summation exceeds a certain threshold, the neuron responds with impulse nerve

(potential action), else if the summation is below the threshold, it remains inactive. The way and the how that human brain handle information is done by acquiring knowledge through learning processes that start from childhood.This biological structure and function was imitated by first Macchllogh and Pitts in 1943[83] where they invented the first formal neuron. In 1950, Rosenblatt created the perceptron endowed with the faculty of learning for classification tasks. In 1960, Muniskey and Perpert added multiple layer of neurons to perceptron to resolve more non-linear problem.

An Artificial Neural Network (ANN) is an imitation of human brain in the structure and also in the way of how to handle information where the structure is imitated by using a set of neurons with multiple layers and for how to handle information, multiples learning algorithm where deployed and classified in general to supervised learning and unsupervised learning, we are interested here in the first way of learning due to our application in text detection. The learning process for ANN is done by presenting an amount of information as input associated with output (supervised learning), where the learning is based on adjusting weight values of ANN in iterative way. These weight are associated to each inputs neurons storing the knowledge learned. The ANN contoured a great success to solve non-linear problem with an efficient generalization and they are more used in image processing domain.

In the following we will give a formal representation of ANN and its learning process. ANN is trained given a set of N training examples where each input $x^i \in X$ is associated with $y^i \in Y$ as label and a number M of iterations (round). As mentioned in Adaboost section we are interested in problem classification of two classes -1 and +1. With hypothesis function based on sigmoid (logistic) activation noted as follow for a single neuron representation (Figure 3.10) with assumption that vector weight associated to this neuron is noted w^T where T is the transpose [84]:

$$h(x^i) = \frac{1}{1 + e^{w^t x^i}} \tag{3.11}$$

To generalize the representation for ANN with multiple layers L (input, hidden and output layers) with multiple neurones m_l in each layers we denote by a_i^j the activation of neuron i in layer j and Θ^j the weights matrix. Therefore given a layer j we denote a variable Z as following :

$$Z^j = \Theta^{j-1} a^{j-1} \tag{3.12}$$

The variable Z is used to calculate the activation of neurons in layer j by applying sigmoid on variable Z^j as follow :

$$h_\Theta(x) = a^j = g(z^j) \tag{3.13}$$

This activation clalculation are propagated from layer 1 to L layer in ANN. Adding all intermediate layers in ANN produce more complex and interesting non-linear hypothesis. After propagation of the values we define the following cost function for learning process:

$$J(\Theta) = \frac{1}{2} \sum_{i=1}^{N} \|\hat{y}_i - y_i\|^2 \tag{3.14}$$

where $\hat{y}_i$ is predicted value for example x_i and is also the associated label to same example. Then, in the following we present a Backpropagation algorithm like the same one used in our application. This algorithm minimize the above cost function

using an optimal set of parameters in theta as following :

given an output y_i associated with input example x_i , we compute the difference error for last layer representing the predicted classe with the correct classification using:

$$\delta^L = a^L - y^i \tag{3.15}$$

Where a^L is the vector of outputs values of the last layer. To compute the delta values of the layers before the last one, we use an equation that steps us back from right to left: therefore to compute δ^{L-1},δ^{L-2}, ...,δ^2 we use.

$$\delta^l = ((\Theta^l)^T \delta^{l+1} * g(z^l) \tag{3.16}$$

Then, the delta values of layer l are calculated by multiplying the delta values in the next layer with the theta matrix of layer l and with the derivative of the activation function g evaluated with the input values given by Z^l :

$$g'(Z^l) = a^l . * (1 - a^l) \tag{3.17}$$

An accumulator noted Δ is used to accumulate back propagated errors to calculate partial derivative of cost function, the value of Δ is given for every neurons in each layer $\Delta^l_{i,j}$ as following with the vectorized form too for each layers.

$$\Delta^l_{i,j} = \Delta^l_{i,j} + a^l_j \delta^{l+1}_i \tag{3.18}$$

$$\Delta^l = \Delta^l + \delta^{l+1}(a^l)^T \tag{3.19}$$

Finally, we compute for back propagation algorithm the partial derivative. As follow.

$$\frac{\partial}{\partial \Theta_{i,j}^{l}} J(\Theta) = \frac{1}{n} \Delta_{i,j}^{l} \tag{3.20}$$

The details of back propagation algorithm is described as following:

Input: Training Data $\{(x^1,y^1),\ldots,(x^N,y^N)\}$
Set $\Delta_{i,j}^{l} = 0$ for all l,i,j.
For i=1 to N do
Set $a^1 = x^1$
Perform forward propagation to compute $a^l$for l=2,3,...,L
Using y^i, compute $\delta^L = a^L - y^i$
Compute δ^{L-1},δ^{L-2}, ...,δ^2
$\Delta_{i,j}^{l} = \Delta_{i,j}^{l} + a_j^l \delta_i^{l+1}$
End for
Compute $\frac{\partial}{\partial \Theta_{i,j}^{l}} J(\Theta) = \frac{1}{n} \Delta_{i,j}^{l}$

3.3.3 Support Vector Machine (SVM):

Another machine learning technique that meet a great success to solve non-linear problems is the Support Vector Machine (SVM) was first introduced in 1992 [85] and was popular after the work of Bottou et al [86] in handwritten digit recognition. The SVM combine many concepts together, using statistical learning theory [87], classification and regression analysis. Therefore, this combination result in a discriminant classifier which distinct the various classes of data by the use of a hyper-plane. SVM model take as input the training data (each example associated with its label classes) and gives as outputs an optimized hyper-plane to separate data

with maximal margin, which lead to reduce the generalization Error and give a very high level of accuracy.

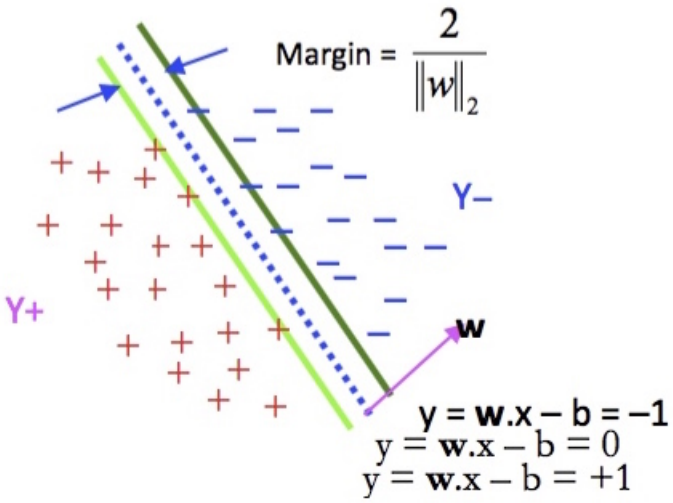

Figure 3.10: The linearly separable problem.

In the Figure 3.10 The Red and Blue are the classes of labelled training data points, classified using linear hyper-plane with optimal margin$=\frac{2}{\|w\|}$. However, the SVM can also be used to classify a non-linear problems using kernels the following Figure 3.11 show the non-linearity problem that can be separated.

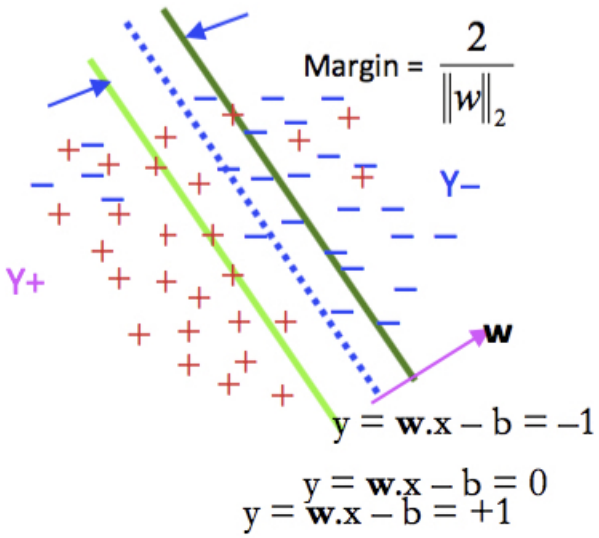

Figure 3.11: The linearly inseparable problem

Given a set of N training examples where each input $x_i \in X$ is associated with $y_i \in Y$ as label with classification problem for two classes -1 and +1 following our text detection systems. With W as coefficients values that must be optimized. Then, the linear separating hyperplane classifier is:

$$f(x) = sgn(w.x - b) \tag{3.21}$$

Therefore, the maximum separating hyperplane margin of two classes HP : y = w.x – b = 0 is calculated with two hyperplanes parallel to it and with equal distances to it given as following,

$$HP1 : y = w.x–b = +1 \; and \; HP2 : y = w.x–b = –1 \tag{3.22}$$

With the condition that distance between HP1 and HP2 is maximized with no data in between. The goal is to maximize the distance between HP1 and HP2 where HP1 contain positive examples and HP2 contain negative examples. These examples are called support vectors because only they participate in the definition of the separating hyperplane [88].

In 2D dimension, the distance from a point (x_0, y_0) to a line $Ax + Bx + C = 0$ is defined as follow:

$$\frac{\mid Ax_0 + By_0 + C \mid}{\sqrt{A^2 + B^2}} \tag{3.23}$$

In the same context, the distance from the hyperplane HP1 to HP : w.x – b = 0 is defined as follow:

$$\frac{\mid w.x - b \mid}{\parallel w \parallel} = \frac{1}{\parallel w \parallel} \tag{3.24}$$

Where the distance between HP1 and HP2 is $\frac{2}{\|w\|}$.

Then, to maximize the distance, we should minimize $\parallel w \parallel = w^T w$ with the condition that there are no data points between HP1 and HP2 $w.x - b \geq +1$, for positive example $y_i = +1$ and $w.x - b \leq -1$, for negative example $y_i = -1$.These two condition can be combined into:$y_i(w.x - b) \geq 1$,. So the problem can be formulated as

$$min_{w,b} \frac{1}{2} w^T w \tag{3.25}$$

subject to $y_i(w.x - b) \geq 1$,

This is a convex, quadratic programming problem where we can use Lagrange multipliers $\alpha_1, \alpha_2 \ldots \alpha_n \geq 0$, in the following way:

$$L\left(w, b, \alpha\right) \equiv \frac{1}{2} w^T w - \sum_{i=1}^{N} \alpha_i y_i \left(w.x_i - b\right) + \sum_{i=1}^{N} \alpha_i \tag{3.26}$$

For non-linear problem like our problem in text detection, the two classes are non-linearly distributed where SVM can transform the data points to another high dimensional space such that the data points will be linearly separable. We define the transformation in high dimensional space as $\Phi(.)$ function. Then we solve the following.

$$L_D \equiv \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \alpha_i \alpha_j y_i y_j \Phi(x_i).\Phi(x_j) \tag{3.27}$$

Suppose, in addition, $K(x_i, y_j) = \Phi(x_i).\Phi(x_j)$. That is, the dot product in that high dimensional space is equivalent to a kernel function of the input space. So, we need not be explicit about the transformation () as long as we know that the kernel function $K(x_i, y_j)$is equivalent to the dot product of some other high dimensional space. The Mercers's condition can be used to determine if a function can be used as a kernel function: There exists a mapping Φand an expansion

$$K(x, y) = \sum_{i} \Phi(x_i).\Phi(x_j) \tag{3.28}$$

if and only if, for any g(x) such that $\int g(x)^2 dx$ is finite, then

$$\int K(x, y) g(x) g(y) dx dy \geq 0 \tag{3.29}$$

The possibility of using different kernels allows viewing learning methods like Radial Basis Function Neural Network (RBFNN) or multi-layer Artificial Neural Networks (ANN) as particular cases of SVM despite the fact that the optimized criteria are not the same. SVM optimizes a geometrical criterion, which is the margin and is sensitive only to the extreme values and not to the distribution of the data into the feature space. The SVM approach transforms data into a feature space F that usually has a huge dimension. It is interesting to note that SVM generalization depends on the geometrical characteristics of the training data, not on the dimensions of the input space. Training a support vector machine (SVM) leads to a quadratic optimization problem with bound constraints and one linear equality constraint. Vapnik shows how training a SVM for the pattern recognition problem leads to the following quadratic optimization problem [88].

Minimize:

$$w(\alpha) = -\sum_{i=1}^{l} \alpha_i + \frac{1}{2} \sum_{i=1}^{l} \sum_{i=1}^{l} y_i y_j \alpha_i \alpha_j k(x_i, x_j) \tag{3.30}$$

Subject to

$$\forall i : 0 \leq \alpha_i \leq C \sum_{i=1}^{l} y_i \alpha_i \tag{3.31}$$

Where l is the number of training examples is a vector of l variables and each component corresponds to a training example (x_i, y_i). The solution of (3.29) is the vector for which (3.29) is minimized and (3.30) is fulfilled.

Chapter 4

IMPLEMENTATION AND RESULTS

In this chapter we present our four implemented systems in details with experiment results done over our data set and also over KIAST dataset to test the segmentation performance of our systems.

4.1 Introduction:

Our proposed system is based on Gomez and Karatzas [17] for detection using gestalt theory. A features set of meaningful group regions is calculated to feed three classifiers SVM, NN, Adaboost. An ensemble method based on majority voting are applied on three outputs to decide which regions are more probable to be text. Then, we did a comparative study with the ensemble method and each three classifiers separately, where ensemble method is found to be the best in precision and accuracy. We also illustrate the results on our own dataset constructed from natural images that contain Arabic script and their ground-truth collected from many sources (indoor, outdoor, book, brochure...) with different sizes and shapes, that can be used for detection and recognition of Arabic Text. Finally we tested our systems on KIAST dataset that contain English and Korean photos with their ground-truth, where the ensemble method shows it efficiency.

4.2 Pipelines of four systems:

The bellow Figure 4.1 represent the pipelines of our three systems where we replace the classification methods by either SVM or NN or Adaboost. For the Ensemble Approaches' the pipeline is different and is shown in the section 3.3.5. The system has as an entry a natural image with Arabic scripts so, we apply Maximal Stable Extremal Region decomposition to get an image region decomposition and then we cluster the meaningful regions using some statistical and geometrical descriptors calculated on those regions and then feed the classifiers with these desciptors. The output of the classifiers will be an image filled with probable regions as text.

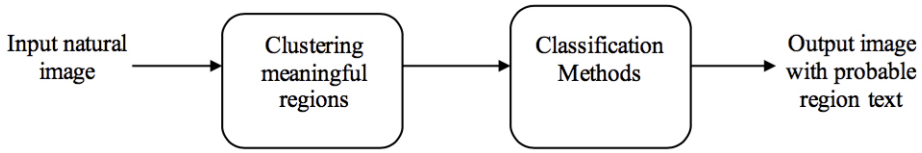

Figure 4.1: Pipelines of our three system SVM, NN and Adaboost

4.3 Machine learning based classifiers and majority voting:

4.3.1 Pre-processing step:

Following cognitive science, the nature of text detection by the human perceptual system is based on perception of atomic objects groups with the same characteristics, this notion leaded to introduce a theory called Gestalt theory. Gomez and Karatzas [17] used a set of hypothesis that compute the perceptual organization through various laws of similarity combined with proximity laws, to have the evidence on the most meaningful group regions. The first step of our work begins with the detection of meaningful regions that can constitute a text [17]. In the first stage of this

detection, a region groups tree is constructed using Maximally Stable Extremal Regions (MSER) decomposition algorithm. In the second stage, a set of features are calculated on group regions of resulting tree where geometrical and statistical features (bounding box area, number of pixels, diameter of the bounding circle, mean intensity value, mean L*a*b* color of the region and its outer boundary) are combined with the coordinates of the regions centers to describe similarity and proximity relations between characters of a word or text line. For each feature sub-spaces, a perceptual organization clustering is applied using bottom-up agglomeration approach to build a set of nodes that constitute a dendrogram. The meaningfulness of each node is tested using Helmholtz principle as probabilistic approach to Gestalt Theory by automatically detecting deviation from randomness corresponding to meaningful events using the trial of binomial distribution as metric. To perform the final clustering analysis of the regions a co-occurrence matrix is used as last stage to accumulate by evidence all maximum meaningfulness clusters of each dendrogram. In the second step, we implemented three classifiers and an Ensemble classifier to filter these meaningfulness clusters as text or non-text.

4.3.2 Support Vector Machine (SVM):

A features set of meaningfulness clusters resulting from accumulating evidence are calculated and fed to SVM classifier. The first classifier is used to classify each region in each group being a character or not, the second one classify a group of regions as text/non-text group. In general the linear separability using hyperplane to separate two classes with important margin is the basic idea of SVM to classify linear problems. For the non-linear problem SVM solve it using kernels that can use high dimensional space to transform data points into others, to make it linearly separable. Some problems like our problem are not linear separable for that we use two SVM's with RBF kernel [14]. The first one was trained to classify regions to be character or not with training set containing examples representing geometrical descriptors. In general, these descriptors define the common nature of characters, that have near similar shapes with near same intensity values in image like (stroke

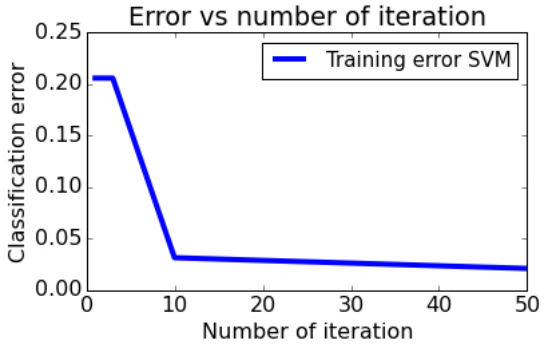

Figure 4.2: Training error for the second SVM

Count	predicted label	Target label
683	1	-1
904	-1	1
60777	-1	-1
15006	1	1

Table 4.1: Confusion matrix for 77370 trained examples

width, area, perimeter, bounding box of the region. numbers of holes, compactness of the region) plus some statistical values on these descriptors like standard deviation, mean of stroke width, aspect ratio between perimeter and area [2]. The second SVM use the label classes and the same descriptors of the first one calculated on text-lines with others statistical values (standard deviation, mean and ratio between them) on each descriptors to classify a region groups to be text or non-text. The two classifiers were trained using our own Dataset, where we extracted a set of characters and text lines. To increase the size of training set, we also added some synthetic words and fonts with certain geometrical transformation in width and high plus dilation, rotation and changing background [3]. Figure 4.2 and Table 4.1 show the training statistics of Second SVM with 98% accuracy.

4.3.3 Neural Network (NN):

In our second approach we keep the same pre-processing model but this time we use two Neural Network classifiers to filter the resulting meaningfulness clusters. As we

discuss in Chapter 3, the Artificial Neural Network is inspired from cognitive science, exactly imitating the human brain where it is modeled as a non-linear computational model based on a set of connected artificial neurons. These neurones are defined by a mathematical function called activation function and connected to each other with weighted values. In the learning process of these model the weights are updated or adjusted following the information passing through and respecting certain learning algorithms[89]. We used a simple NN with backpropagation learning algorithm and sigmoid activation function. Therefore the inputs of the first classifier are the same set of features described in SVM to classify the region as character or not. The output of this classifier is used with the same set of features described by SVM to classify the meaningful region groups as text or non-text. The training accuracy of second NN was 98% . The Figure 4.3 and Table 4.2 illustrate the training errors and confusion matrix.

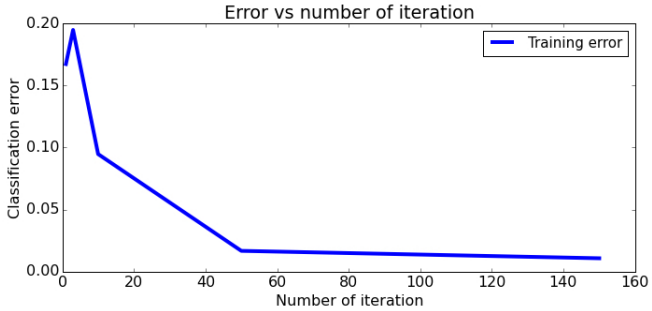

Figure 4.3: Training error for the second NN

Count	Predicted label	Target label
61393	-1	
2661	-1	-1
67	1	1
13249	1	1

Table 4.2: Confusion matrix for 77370

4.3.4 Adaboost:

Adaboost is a strong classifier based on the combination of weighted week learners that gives a set of hypotheses during the learning processes [90]. Adaboost then try to find a linear combination of these weighted week classifiers to produce a strong hypothesis about the classification problem. With Adaboost the training was very fast with an accuracy of 100%. Figure 4.4 and Table 4.3 shows the training error and confusion matrix respectively.

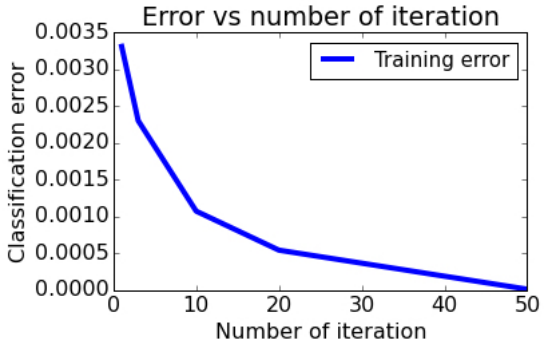

Figure 4.4: Training error for the second Adaboost classifiers

Count	Predicted label	Target label
61460	-1	-1
15910	1	1

Table 4.3: Confusion matrix for 77370 trained examples

4.3.5 Ensemble Approach (EA):

An ensemble classifier approach shows a great improvement by combining a set of classifiers (base learners) to classify a set of data using majority voting or averaging [91]. In our work, we opted for majority voting class to decide on the final output of ensemble classifiers (SVM, Adaboost, NN). Figure 4.5 show the architecture of architecture of the ensemble model. The idea is that the meaningful group resulted from clustering stage is presented to the three group classifiers where each classifier give a class for this region and the the major class presented as output from the three

classes is chosen. For example, if SVM and Adaboost vote positive for the text region and NN (negative) the final outcome will be decided as positive.

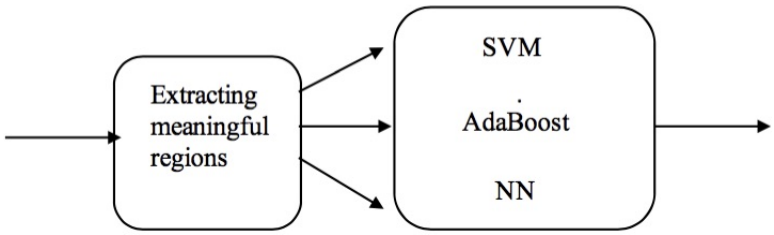

Figure 4.5: Pipeline of ensemble approach.

4.4 Results:

We constructed our database set of 300 images taken from indoor, outdoor and books most of them containing Arabic text with few English text too. From this database, we extracted a training set for character and text classifiers. We expanded this training set by adding transformed synthetic fonts and words for more accuracy in the training stage of the system. The size of training set for character classifier was 16037 examples containing most Arabic characters and few English letters with important negative examples representing non-character images. For the training set of text regions, the size of dataset was 77370 examples containing Arabic and few English text lines images too with negative examples as non-text regions. To asses the performance of the developed systems we calculated for each one the precision, recall, and accuracy at pixels' images stage, where we use ground-truth images for every image presented for detection. The precision is defined as the ratio between the sum of pixels correctly detected as text and the total number of pixels detected as text. Recall is defined as the ratio between the sum of text pixels correctly detected and the sum of actual text pixels in ground-truth image. Finally the accuracy is defined as sum of pixels correctly detected as text plus sum of pixels correctly detected as background over the sum of all pixels in image. Table 4.4 shows that the performance of the ensemble method, which is better than others with 96% accuracy and 60% precision. If we

compare the three classifiers without ensemble, we find that Adaboost performance is better with 52% in precision and 95% in accuracy. Figure 4.6 shows the comparative performance and Table 4.4 illustrate the results of text region detection for two images from the Dataset.

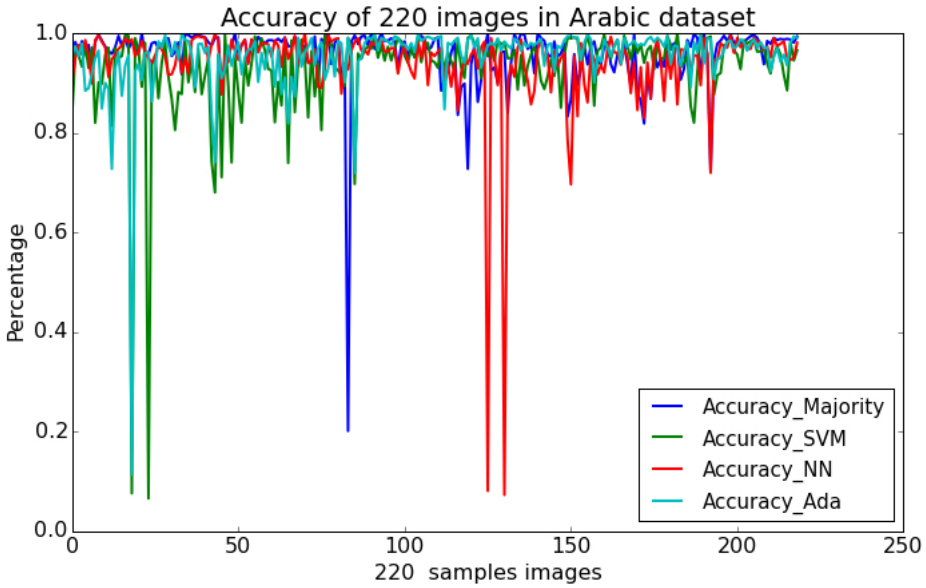

Figure 4.6: Comparative accuracy graph between all classifiers

	Precision	Recall	Accuracy
Ada-boost	52%	32%	95%
SVM	47%	59%	93%
NN	45%	59%	93%
Ensemble	60%	53%	96%

Table 4.4: Test on Arabic Dataset with 220 images

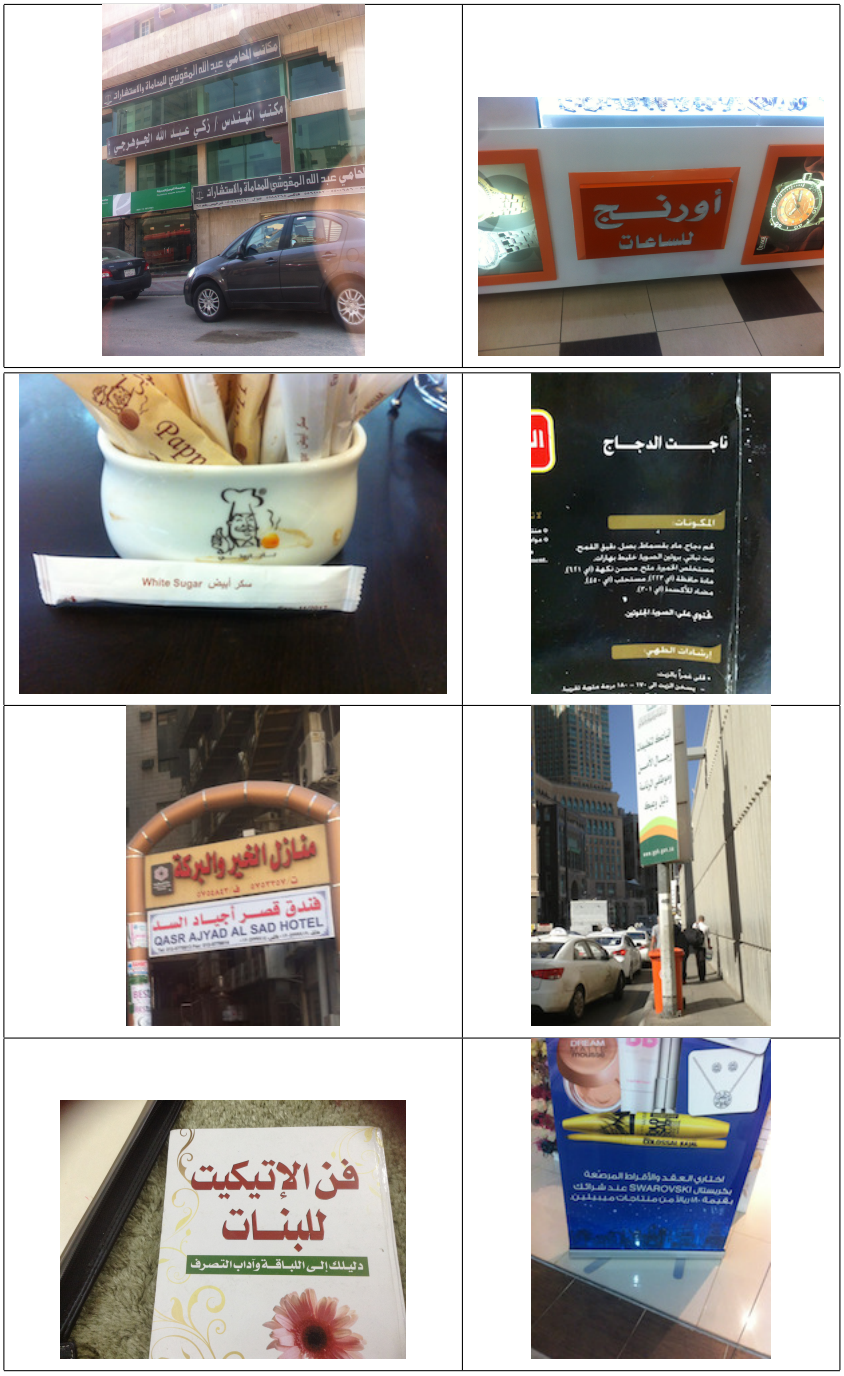

Figure 4.7: Original images from our Dataset.

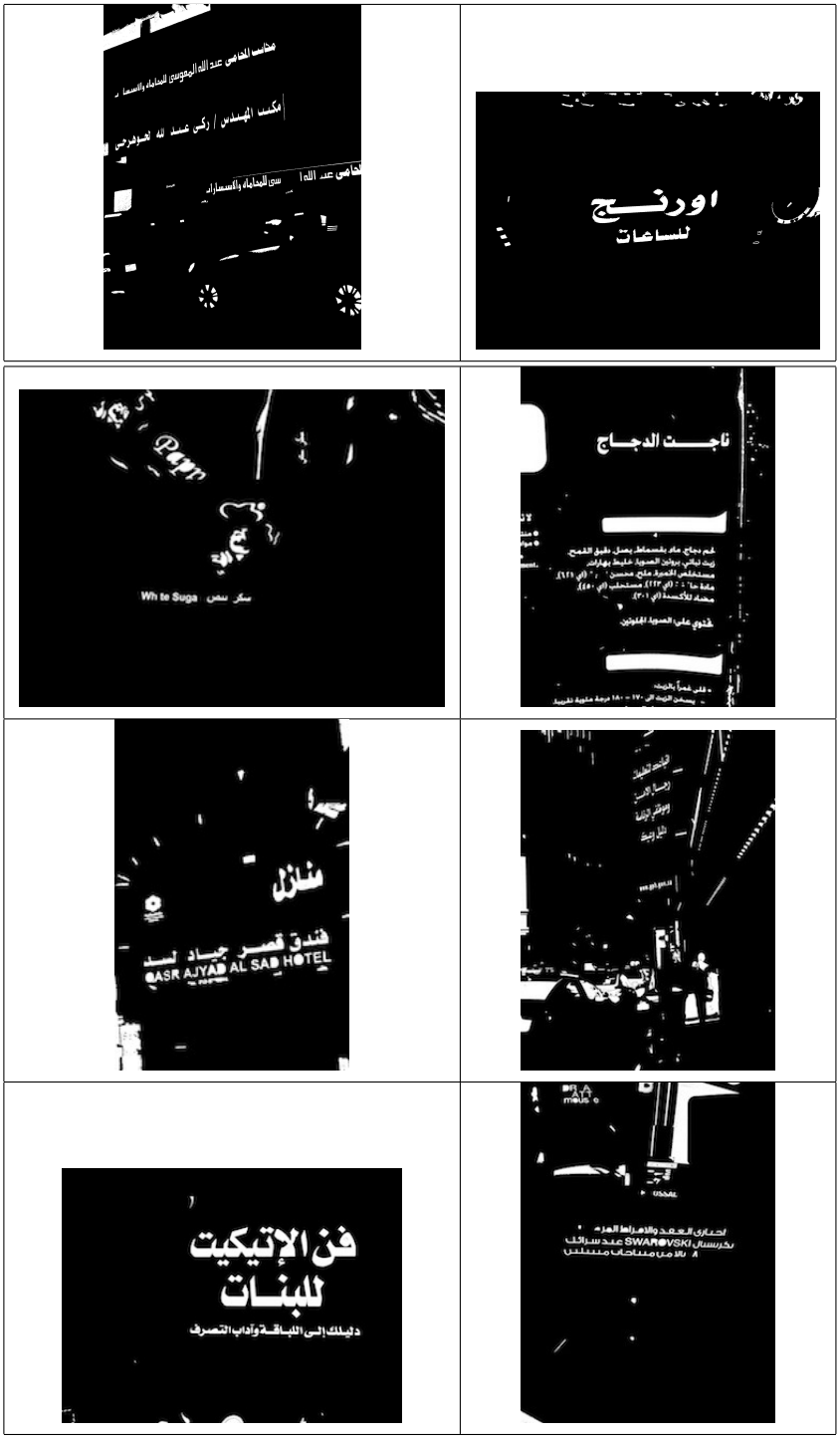

Figure 4.8: Results of text regions detection with SVM classifier.

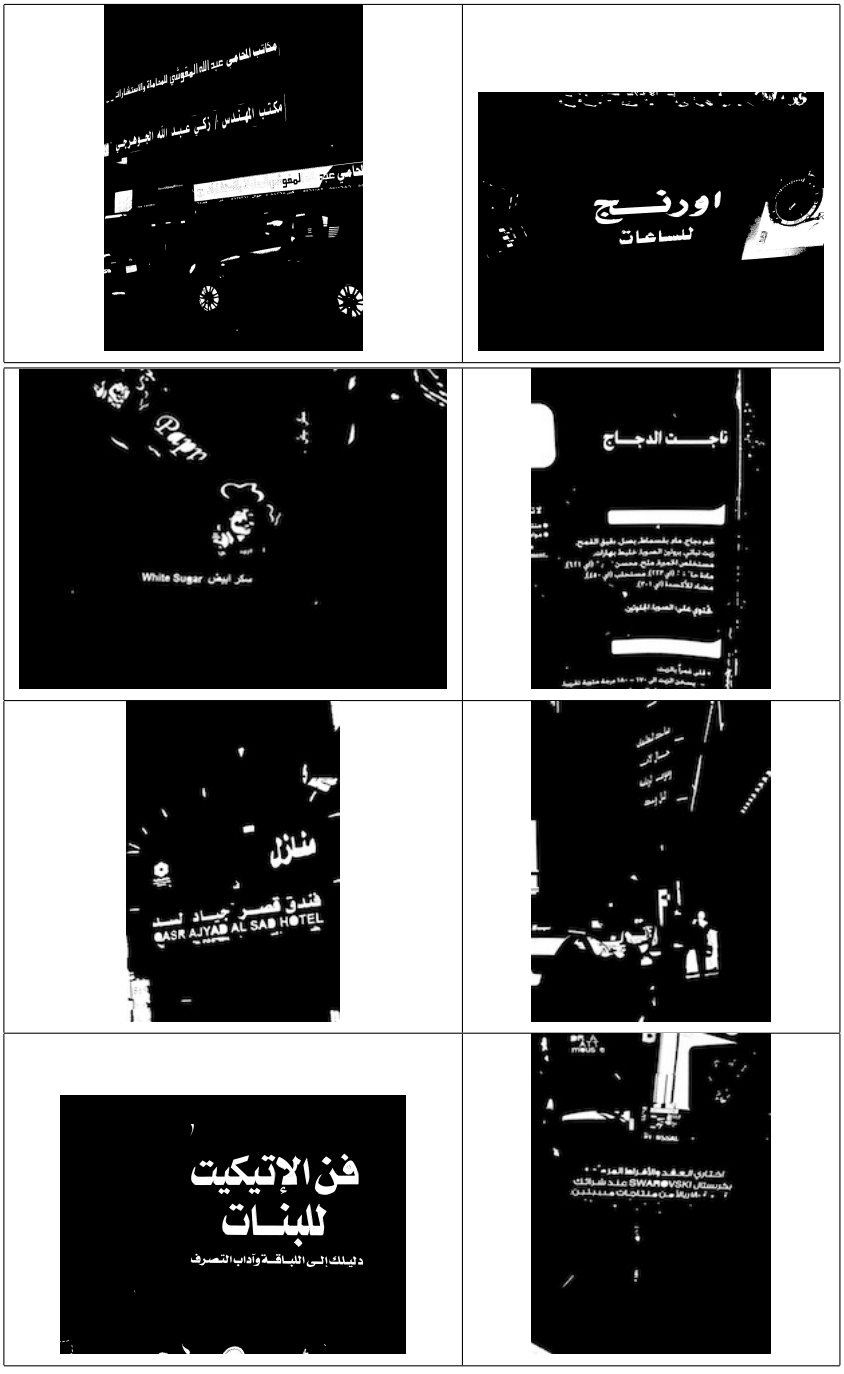

Figure 4.9: Results of text regions detection with NN classifier.

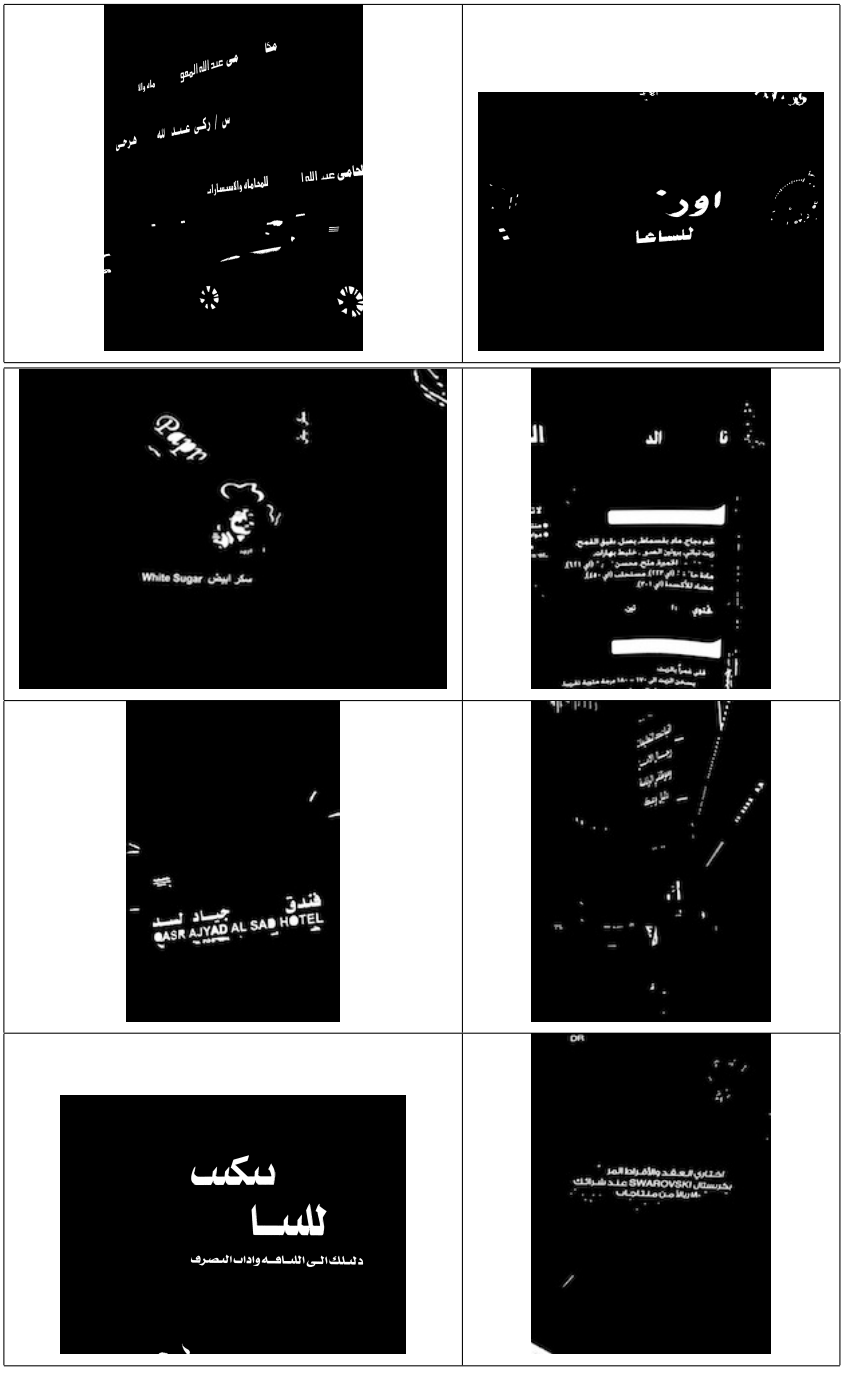

Figure 4.10: Results of text regions detection with Adaboost classifier.

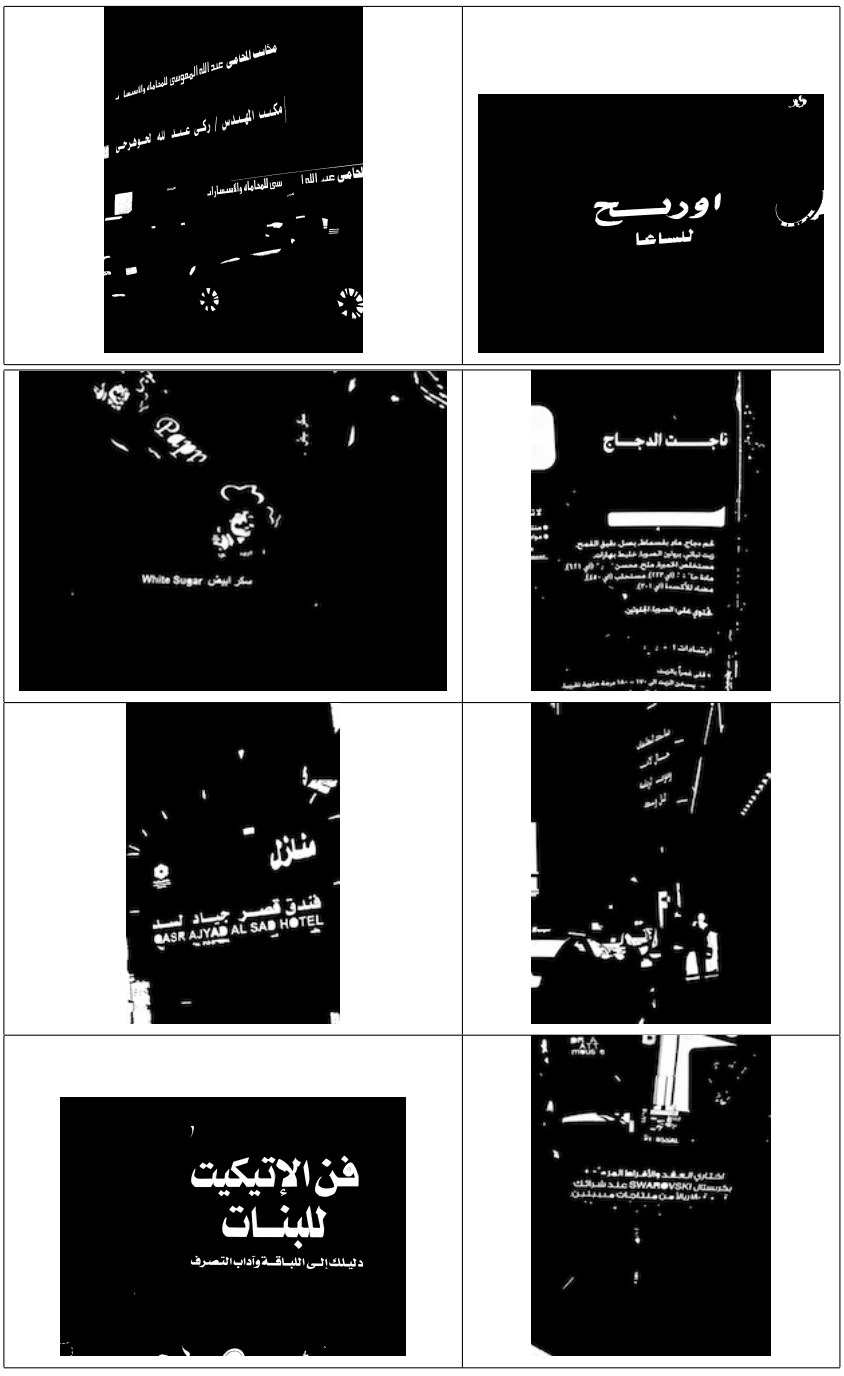

Figure 4.11: Results of text regions detection with Ensemble approach.

We tested our four classifiers on KAIST dataset [93] as segmentation level test. The KAIST dataset consisting on 3000 natural scene images, Those images contain most Korean languages and in the remaining images we found Mixed ones (Korean and English). For our experiments we use 2000 images corresponding to all kind of images of KAIST dataset. Table 4.5 blow show the obtained results on the KAIST dataset.

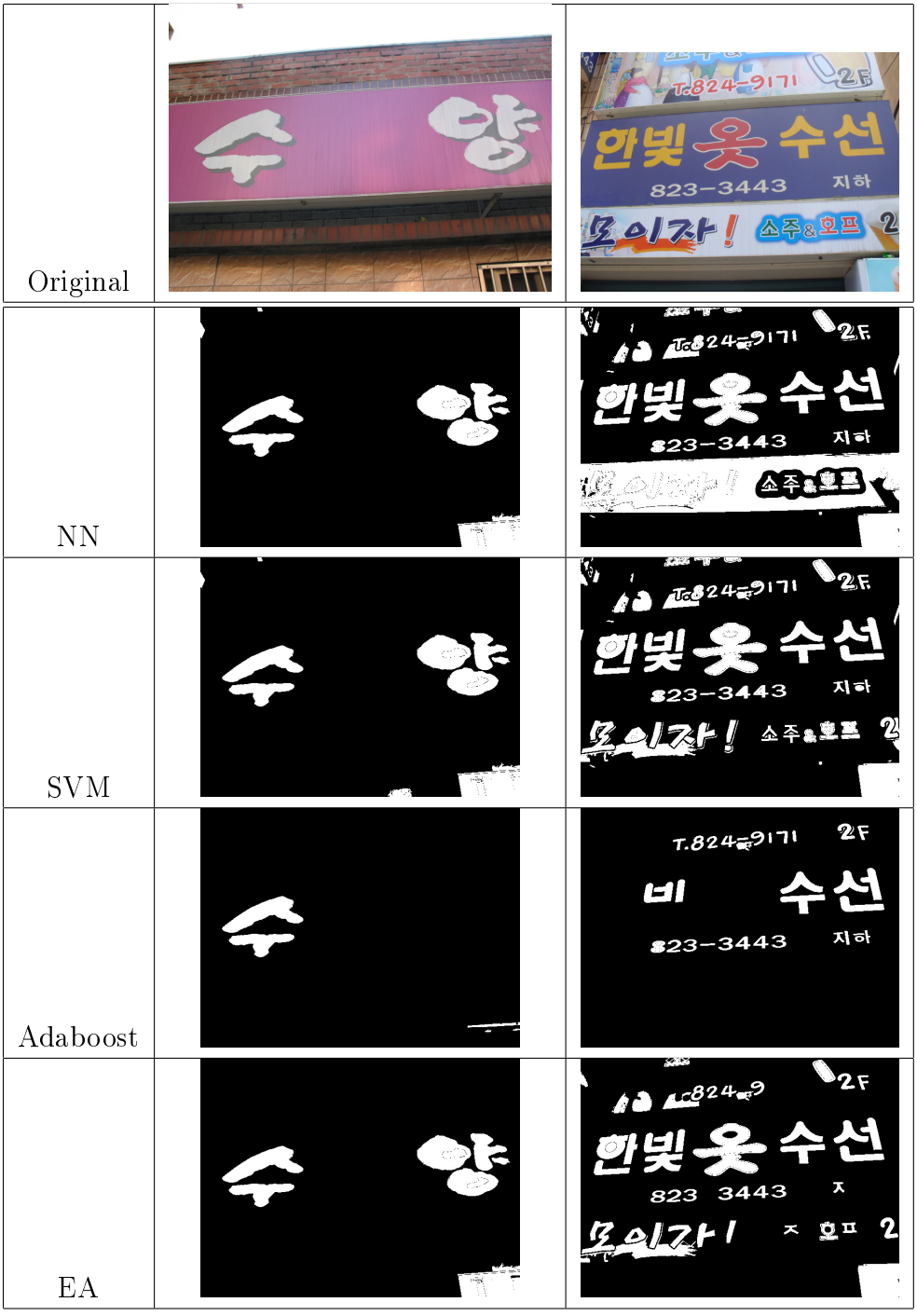

Table 4.5: Results of text regions segmentation with all classifiers on KAIST examples.

CONCLUSION:

This thesis addresses the problem of Arabic text detection by proposing four systems based on machine learning to solve the problem in natural images. Our goal was to cover the lack of the existing of such system for Arabic language also the lack of the existing of database containing natural image with Arabic text script. In our work, we constructed first a database containing natural images with Arabic text regions taken from different sources with complex background and under different natural condition like blurring, low resolution and distortion..etc. Next, we implemented four systems that can detect Arabic text regions with different size, color and orientation.

Our contribution is the combination of probabilistic method with machine learning methods to solve the detection problem. The first method use Gestalt theory based on set of proximities and similarities associated with hierarchical clustering to gather all meaningful regions probable to be text. Then, an ensemble of machine learning methods were implemented to classify the meaningful regions to text or non-text. For that, a set of features were calculated on these meaningful regions and presented as input for the four classifiers. The first one used SVM that was better in recall, the second one used NN and the third one was Adadboost where it gave a better precision and accuracy. For the last one the ensemble approach combined the three classifiers where the performance was the best over all. As major advantage of using perception principal of Gestalt theory is that the resulted group regions are not restricted to a specific kind of size, orientation..etc. This invariance helps to use our four systems as segmentation systems also. Therefore, we performed some experiments on KAIST database to test the degree of segmentation of our system where ensemble approach

gave better performance.

The perspectives of our work will be introducing words level recognition using the input regions detected as text to construct end-to-end system detection and recognition of Arabic script. Also we can improve and enhance the detection step by introducing the localization notion using the bounding boxes. Also, associate an annotation file to each image in our database to be able to test the performance of our systems with our database and others database available online like ALIF, AcTiV... etc.

REFERENCES

1. X.-C. Yin, X. Yin, K. Huang, and H.-W. Hao. Robust text detection in natural scene images. In IEEE Transactions on Pattern Analysis and Machine Intelligence, pages 970–983, 2014

2. Z. Zhao, C. Fang, Z. Lin, and Y. Wu. A Robust Hybrid Method for Text Detection in Natural Scenes by Learning- based Partial Differential Equations, Neurocomputing, Vol. 168, pp. 23-34, 2015

3. M. M. Grond. Text detection in natural images using convolutional neural networks. Thesis, Stellenbosch University, 2017.

4. S. Yousfi. Embedded Arabic text detection and recognition in videos. Document and Text Processing. PHD Thesis. Université de Lyon, 2017.

5. A. Bissacco, M. Cummins, Y. Netzer, and H. Neven. Photoocr: Reading text in uncontrolled conditions. In ICCV, 2013.

6. B. Epshtein, E. Ofek, and Y.Wexler. Detecting text in natural scenes with stroke width transform. In CVPR, 2010.

7. J. Canny. A Computational Approach To Edge Detection. IEEE Trans. Pattern Analysis and Machine Intelligence, 8:679-714, 1986.

8. B. K. P. Horn. Robot Vision. McGraw-Hill Book Company, New York, 1986.

9. Cong Yao, Xiang Bai, Wenyu Liu, Yi Ma, and Zhuowen Tu. Detecting texts of arbitrary orientations in natural images. CVPR'12, pp. 1083–1090, 2012.

10. T. Hastie, R. Tibshirani, and J. Friedman. The elements of statistical learning: Data mining, inference, and prediction, second edition. New York: Springer, 2009.

11. L. Neumann and J. Matas. Real-time scene text localization and recognition. In CVPR 2012, pages 3538 –3545, 2012.

12. L. Neumann and J. Matas. Text localization in real-world images using efficiently pruned exhaustive search. In ICDAR2011, pages 687–691, 2011.
13. J. Matas and K. Zimmermann. A new class of learnable detectors for categorisation. In Image Analysis, volume 3540 of LNCS, pages 541–550. 2005.
14. K.-R. Muller, S. Mika, G. Ratsch, K. Tsuda, and B. Scholkopf. An introduction to kernel-based learning algorithms. IEEE Trans. on Neural Networks, 12:181–201, 2001.
15. J. Matas, O.Chum, M.Urban, and T.Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In Proc. BMVC., pages 384–393, 2002.
16. M. Kuchaki Rafsanjani, Z. A. Varzaneh, N. E. Chukanlo. A survey of hierarchical clustering algorithms. The. Journal of Mathematics and Computer Science Vol .5 No.3, pp 229-240, 2012.
17. L. Gomez and D. Karatzas. Multi-script text extraction from Natural scenes. in ICDAR, 2013.
18. A. Desolneux, L. Moisan, and J.-M. Morel. A grouping principle and four applications. IEEE Trans. PAMI, 2003.
19. A. Fred and A. Jain. Combining multiple clusterings using evidence accumulation. IEEE Trans. PAMI, 2005.
20. S. M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, and R. Young. ICDAR 2003 robust reading competitions: entries, results, and future Directions. IJDAR, 2005.
21. H. Gaddour, S. Kanoun and N. Vincent. A New Method for Arabic Text Detection in Natural Scene Image based on the color homogeneity. International Conference on Image and Signal Processing (ICISP), 2016
22. H. Gaddour, S. Kanoun and N. Vincent . Color Stability and Homogeneity Regions to Detect Text in Real Scene Images. CSHR. ICDAR, 2017.
23. W. Ding, S. Shan and F. Su. Text detection in natural scene images by hierarchical localization and growing of textual components. 2017 IEEE International Conference on Multimedia and Expo (ICME), Hong Kong, Hong Kong, pp. 775-780, 2017.
24. M. Halima, H. Karray and A. Alimi. Arabic Text Recognition in Video Sequences.

in Proceeding of International Conference on Informatics, Cybernetics and Computer Applications, Bangalore, pp. 603-608, 2010.
25. M. Moradi, S. Mozaffari. Hybrid Approach for Farsi/Arabic Text Detection and Localisation in Video Frames. Processing.7(2), 2013.
26. X. Chen, A. Yuille. Detecting and Reading Text in Natural Scenes. Computer Vision and Pattern Recognition (CVPR), pp. 366-373, 2004.
27. F. Slimane, R. Ingold, M. A. Alimi and J. Hennebert. Duration Models for Arabic Text Recognition using Hidden Markov Models. CIMCA 2008, Vienne, Austria, 2008.
28. M S. Khorsheed. Offline recognition of omnifont Arabic text using the HMM ToolKit (HTK). Pattern Recognition Letters 28(12), pp. 1563-1571, 2007.
29. H. Li, D. Doermann and O. Kia. Automatic text detection and tracking in digital video. IEEE Transactions on Image Processing, vol. 9, no. 1, pp. 147–156, 2000.
30. R. Lienhart and A. Wernicke. Localizing and segmenting text in images and videos. IEEE Transactions on Circuits and Systems for Video Technology. vol. 12, no. 4, pp. 256–268, 2002.
31. M. Jain, M. Mathew and C.V. Jawahar. Unconstrained Scene Text and Video Text Recognition for Arabic Script. ASAR 2017.
32. S. Yousfi, S.-A. Berrani, C. Garcia. Arabic text detection in videos using neural and boosting-based approaches: application to video indexing. In International Conference on Image Processing, Paris, France, pp. 3028-3032, 2014.
33. M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman. Synthetic data and artificial neural networks for natural scene text recognition. Proc. NIPS Workshops, 2014.
34. L. Zhang, R. Chu, S. Xiang, S. Liao, and S.Z Li. Face detection based on multi-block lbp representation. in Proceedings of the international conference on Advances in Biometrics, Seoul, Korea , August 2007.
35. Lim, Y.K., Choi, S.H., Lee, S.W. Text extraction in MPEG compressed video for content-based indexing. Proc. 15th Int. Conf. Pattern Recognition, Barcelona, Spain, vol. 4, pp. 409–412, 2000.
36. Y. Zhong, H. Zhang, A.K Jain. Automatic caption localization in compressed

video'. IEEE Int. Conf. Image Processing. vol. 2, pp. 96–100, 1999.
37. A. Shahab, F. Shafait , A. Dengel. ICDAR2011 robust reading competition challenge2 : reading text in scene images. in: ICDAR, pp. 1491–1496, 2011.
38. R. Liu, Z. Lin, W. Zhang, K. Tang, Z. Su. Toward designing intelligent PDEs for computer vision: an optimal control approach. Image Vis. Comput. 43–56, 2013.
39. R. Liu, J. Cao, Z. Lin, S. Shan. Adaptive partial differential equation learning for visual saliency detection. in: CVPR, IEEE, pp.3866–3873, 2014.
40. Y.F. Pan, X. Hou, C.L. Liu. Text localization in natural scene images based on conditional random field. in: International Conference on Document Analysis and Recognition, pp. 6–10, 2009.
41. D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. Gomez, S.R. Mestre, J. Mas, D.F. Mota, J.A. Almazan, L.P. de las Heras. ICDAR 2013 robust reading competition, in Proceedings of the 12th International Conference on Document Analysis and Recognition ICDAR 2013. pp. 1484–1493, 2013.
42. H. Al-Muhtaseb, S. Mahmoud, R. Qahwaji .A novel minimal Arabic script for preparing databases and benchmarks for Arabic text recognition research. Paper presented at the 8th WSEAS International Conference on Signal Processing (SIP), May 30- June 1, 2009.
43. T. E. de Campos, B. R. Babu, and M. Varma. Character recognition in natural images. in Proc. of VISAPP, 2009.
44. A. Mishra, K. Alahari, and C. V. Jawahar. Scene text recognition using higher order language priors. in Proc. of BMVC, 2012.
45. K. Wang and S. Belongie. Word Spotting in the Wild. The 11th European Conference on Computer Vision (ECCV), September 2010.
46. S. Yousfi, S. Berrani, and C. Garcia. ALIF: A dataset for Arabic embedded text recognition in TV broadcast. Document Abalysis and Recognition (ICDAR), 2015 13th international Conference on. IEEE, 2015.
47. O. Zayene, J. Hennebert, S.M. Touj, R. Ingold, N.E. Amara. A dataset for Arabic text detection tracking and recognition in news videos-AcTiV. ICDAR 2015. 2015
48. D. Karatzas, L. Gomez-Bigorda,A. Nicolaou, S. Ghosh,A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V.R. Chandrasekhar, S. Lu, et al. ICDAR 2015 competition

on robust reading. In Proceedings of the 2015 13th International Conference on Document Analysis and Recognition (ICDAR), Tunis, Tunisia, 23–26 August 2015; pp. 1156–1160.

49. M. Tounsi, I. Moalla, M. Alimi. ARASTI: A Database for Arabic Scene Text Recognition. 2017 IEEE International Workshop on Arabic Script Analysis and Recognition (ASAR). 2017

50. F. Slimane, R. Ingold, S. Kanoun, M. A. Alimi and J. Hennebert. Database and Evaluation Protocols for Arabic Printed Text Recognition. Internal research report, DIUF, University of Fribourg, Switzerland, 2009.

51. V. Märgner, H. El Abed, M. Pechwitz. Offline Handwritten Arabic Word Recognition Using HMM - a Character Based Approach without Explicit Segmentation. in the 9th Colloque International Francophone sur l'Ecrit et le Document , CIFED 2006, Sep. 18-21, 2006.

52. L. M. Lorigo, and V. Govindaraju. Offiline Arabic handwritting recognition: a survy. IEEE transactions on pattern analysis and machine intelligence 28.5 (2006): 712-724.

53. http://www.ifnenit.com/ . IFN/ENIT-database –Database Of Handwritten Arabic Words. 2006.

54. M. Pechwitz and M. Volker. HMM Based Approach for Handwritten Arabic Word Recognition Using the IFN/ENIT- Database. The Seventh International Conference on Document Analysis and Recognition (ICDAR 2003), pp. 890-894, 2003.

55. S. Al-Ma'adeed, D. Elliman and C.A. Higgins "A Database for Arabic Handwritten Text Recognition Research," The Eighth International Workshop on Frontiers in Handwriting Recognition (IWFHR'02), 2002.

56. Y.A. Alotaibi. High Performance Arabic Digits Recognizer Using Neural Networks. The International Joint Conference On Neural Networks 2003, pp. 670-674, 2003.

57. M. soleymani, F. Razzazi. An Efficient Front-End system for Isolated Persian/Arabic Character Recognition of Handwritten Data-Entry Forms. pp. 6 2003.

58. O. Zayene , S. Masmoudi Touj , J. Hennebert, R. Ingold, N.E. Amara. Open Datasets and Tools for Arabic Text Detection and Recognition in News Video Frames.

Imaging Journal 2018. pp. 4- 32, 2018.
59. Y. Al-Ohali, M. Cheriet and C. Suen. Databases for recognition of handwritten Arabic cheques. Pattern Recognition, vol. 2003, no. 36, pp. 111-121, 2003.
60. S.S. Maddouri, H. Amiri, A. Belaid and C. Choisy. Combination of local and global vision modelling for arabic handwritten words recognition. Frontiers in Handwriting Recognition, pp.128-135, 2002.
61. V. Margner and M. Pechwitz. Synthetic Data for Arabic OCR System Development. The 6th International Conference on Document Analysis and Recognition, ICDAR'01, pp. 1159-1163, 2001.
62. A. Hamid and R. Haraty. A Neuro-Heuristic Approach for Segmenting Handwritten Arabic Text. ACS/IEEE International Conference on Computer Systems and Applications (AICCSA'01), pp. 110-113, 2001.
63. M. Dehghan, K. Faez, M. Ahmadi and M. Shridhar. Handwritten Farsi (Arabic) word recognition: a holistic approach using discrete HMM. Pattern Recognition, vol. 2001, no. 34, pp. 1057-1065, 2001.
64. N. Kharma, M. Ahmed and R. Ward. A New Comprehensive Database of Hadritten ArabicWords, Numbers, and Signatures used for OCR Testing. 1999 IEEE Canadian Conference on Electrical and Computer Engineering, pp. 766-768, 1999.
65. I. Bazzi, C. LaPre, J. Makhoul, C. Raphael and R. Schwartz. Omnifont and Unlimited-Vocabulary OCR for English and Arabic. 4th International Conference Document Analysis and Recognition (ICDAR '97), pp. 842-846, 1997.
66. B. Bataineh. A Printed PAW Image Database of Arabic Language for Document Analysis and Recognition. J. ICT Res. Appl., Vol. 11, pp 200-212, No. 2, 2017.
67. E. Fernandez-Moral, R. Martins, D. Wolf, P. Rives. A new metric for evaluating semantic segmentation: leveraging global and contour accuracy. Workshop on Planning, Perception and Navigation for Intelligent Vehicles, PPNIV17, Vancouver, Canada, Sep 2017.
68. S. Kolkur, D. Kalbande, P. Shimpi, C. Bapat, J. Jatakia. Human Skin Detection Using RGB, HSV and YCbCr Color Models. CoRR abs/1708.02694, 2017.
69. T. Mitchell. Machine learning. McGraw Hill, New York. ISBN 0-07-042807-7, 1997.

70. G. Kanizsa. Grammatica del Vedere La Grammaire du Voir. Il Mulino, Bologna Editions Diderot, Arts et Sciences, 1980 / 1997.

71. S.-C. Zhu,.Embedding gestalt laws in markov random elds. IEEE Transactions on Pattern Analysis and Machine Intelligence , vol. 21, pp. 1170-1187, 1999.

72. A. Desolneux, L. Moisan, and J.-M. Morel. A grouping principle and four applications. IEEE Transactions on Pattern Analysis and Machine Intelligence , vol. 25, pp. 508-513, 2003.

73. J. H. Elder and R. M. Goldberg. Ecological statistics of Gestalt laws for the perceptual organization of contours. Journal of Vision, 2(4):324-353, 2002.

74. A.J. Bell and T.J. Sejnowski. Edges are the independent components of natural scenes. Advances in Neural Information Processing Systems, 9, 1996.

75. H. von Helmholtz. Treatise on Physiological Optics. Thoemmes Press, 1999.

76. F. Attneave. Some informational aspects of visual perception. Psychology Review, 61:183-193, 1954.

77. L. Gomez, D. Karatzas. Perceptual Organization for Text Extraction in Natural Scenes. Report of the Master Project, Universitat Autònoma de Barcelona, 2013.

78. A. Fred and A. Jain. Data clustering using evidence accumulation, in Pattern Recognition. 2002. Proceedings. 16th International Conference on , vol. 4, pp. 276-280 vol.4, 2002.

79. Y. Freund and R. E. Schapire. Game theory. on-line prediction and boosting. In Conference on Computational Learning Theory. pages 325–332. ACM, 1996.

80. Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Sciences, 55 (1):119–139, 1997.

81. N. Nikolaou.Cost-Sensitive Boosting A Unified Approach. PHD Thesis , Faculty Of Science & Engineering, University Of Manchester, 2016.

82. R. E. Schapire and Y. Singer. Improved boosting algorithms using confidencerated predictions. Machine Learning, 37 (3):297–336, 1999.Simon Haykin and Neural Network. A comprehensive foundation. Neural Networks, 2004.

83. W. S. McCulloch, W. H Pitts. A Logical Calculus of the Ideas Immanent in nervous Activity. Bulletin of Mathematical Biophysics 5(1943), 115 – 133, 1943.

84. A. Ng. Machine Learning Course. Stanford University, 2017.
85. B.E. Boser et al. A Training Algorithm for Optimal Margin Classifiers. Proceedings of the Fifth Annual Workshop on Computational Learning Theory 5 144-152, Pittsburgh, 1992.
86. L. Bottou et al. Comparison of classifier methods: a case study in handwritten digit recognition. Proceedings of the 12th IAPR International Conference on Pattern Recognition, vol. 2, pp. 77-82.
87. V. Vapnik. The Nature of statistical learning theory. Book, January 1995.
88. A. Ajith and N. Sajith and P. P. Sarathchandran. Modelling chaotic behaviour of stock indices using Intelligent Paradigms. Neural, Parallel. & Scientific Computations Archive, 11, 143-160, 2003.
89. G. Zhang. Neural networks for classification: a survey. IEEE Transactions on Systems, Man, and Cybernetics. Part C 30(4): 451- 462, 2000.
90. V. Nikulin, G. McLachlan, S. Ng. Ensemble Approach for Classification of Imbalanced Data. Proceedings of the 22nd Australian Joint Conference on Advances in Artificial Intelligence, Springer-Verlag, 2009.
91. W. Wang. Some fundamental issues in ensemble methods. In World Congress on Computational Intelligence, Hong Kong, pp. 2244–2251. IEEE, Los Alamitos, 2008.
92. F. Gasparini, S. Corchs, R. Schettini. Recall or precision-oriented strategies for binary classification of skin pixels. journal of Electronic Imaging 17(2), 023017, 2008.
93. S. Lee, M. S. Cho, K. Jung, and J. H. Kim. Scene text extraction with edge constraint and text collinearity. in Proc. ICPR, 2010.

Abstract

The automatic detection and recognition of zone text in natural images remain indispensable due to the omnipresent of text information in daily human life. This domain contoured a development of many applications specially with English language where many systems were implemented and proved their efficiency. Arabic language represents a real challenge for its cursive nature and rich vocabulary. The first step of our work was inspired from Gomez and Karatzas [17] on multiscript detection using Gestalt theory. For the second step, we implemented three classifiers namely Neural Network, Support Vector machine and Adaboost. These classifiers were deployed to classify the group regions in images as text or non-text. To improve the system performance an ensemble method based on majority voting was applied where the outputs of the three classifiers were fused. Experiments were conducted using own image database and ground-truth and the empirical results illustrate that the proposed method is efficient.

Résumé

La détection et la reconnaissance automatique des zones de texte dans des images naturelles est indispensables en raison de l'omniprésence de l'information textuelle dans la vie quotidienne des êtres humains. Ce domaine a contourné un développement de nombreuses applications spécialement avec la langue anglaise où de nombreux systèmes ont été mis en œuvre et ont prouvé leur efficacité. La langue arabe représente un véritable défi pour sa nature cursive et son riche vocabulaire. La première étape de notre travail a été inspirée du travail de Gomez et Karatzas [17] sur la détection multiscript en utilisant la théorie de Gestalt. Pour la deuxième étape, nous avons implémenté trois classifieurs: Neural Network, Support Vector Machine et Adaboost. Ces classifieurs ont été déployés pour classer les régions groupées par la premier étape en texte ou non-texte. Pour améliorer les performances des systèmes, une méthode d'ensemble basée sur le vote de la majorité a été appliquée sur les résultats des trois classifieurs. Les expériences ont été menées en utilisant notre propre base de données d'images où les résultats empiriques illustrent que la méthode proposée est efficace.

ملخص

الاكتشاف و التعرف الآلي للمنطقة النصية في الصور الطبيعية يعتبر أساسي بسبب وجود الدائم للمعلومات النصية في الحياة اليومية للإنسان . هذا المجال يعرف تطورا ملحوظا في العديد من التطبيقات الذكية خاصة مع اللغة الانجليزية حيث تم تطوير العديد من الأنظمة التي اتبثت كفاءتها . أما بالنسبة للغة العربية فهي تمثل تحدي حقيقي بسبب طبيعتها المتداخلة في الكتابة ومفرداتها الغنية . في عملنا هذا تم استلهام الخطوة الأولى من عمل لويس[١٧] لاكتشاف المناطق المتشابهة والممكن تواجد النص فيها وفي الخطوة الثانية تم استخدام ثلاث مصنفات وهي الشبكات العصبية الاصطناعية والخورزميات الخطية وادبست . هذه المصنفات تم استعمالها لتصنيف المناطق النصية من الغير نصية في الصور الطبيعة . لتحسين اداء النظام تم دمج مخرجات النتائج الثلاث الخاصة بالمصنفات و تجميعها في واحد وذلك استنادا الى اعلى نسبة من المخرجات . تم تقييم نتائج التطبيقات باستخدام قاعدة بيانات خاصة بنا حيث اثبتث طريقة التجميع فعاليته .