Republic of Iraq Ministry of Higher Education and Scientific Research

University of Technology

Computer Sciences Department

Analysis and Segmentation of Digital Video to Detect Script Writings

A Thesis Submitted to the Department of Computer sciences at the University of Technology as a partial fulfillment of the requirements for the degree of Master of software in Computer Sciences

BY

Teaba Wala Aldeen Khairi

Supervised By

Prof. Dr. Abdul Monem Saleh Rahma

June- 2015

صدق الله العظيم

سورة الفاتحة

Acknowledgments

I would like to express my sincere gratitude and appreciation to:

My supervisor prof. ***Dr. Abdul Monem S. Rahma*** *for his support, advice, and kindness which encouraged me to finish this work.*

My husband ***Dr. Mohammed Moanes Ezzaldean*** *for his help, and love which gave me great support all the time.*

My ***Father*** *and* ***Mother****, for there great love, support, help, and assistant.*

My friend ***Raghad Kamil*** *for her real friendship, and helpful efforts all the time.*

Dedication

To my beloved husband ….

With whose support and encouragement I could pass the difficulties….

To the kind heart and hidden pearls …

The most beloved who loves and encourages me …

Dear Mother and Father…

To my lovely daughters and son Joumnah, Jannah, yousif

To the second father, the late Dr. Emad Kadhim.

To friends and family who support me, especially Raghad .

Teaba

Examination Committee Certificate

We certify that we have read the thesis entitled "Analysis and Segmentation of Digital Video to Detect Script Writings" and as an examination committee, examined the student "Teaba Wala Aldeen Khairi " in its content and that in our opinion, it meets the standard of a thesis for the degree of Master in computer sciences

Signature:

Name: Asst. Prof. Dr. Amer S. AL-Mallah (chairman)

Date:

Signature:

Name: Asst. Prof. Dr. Hanaa Mohsin Ahmed (Member)

Date: / /2015

Signature:

Name: Asst. Prof. Dr. Yossra Hussain Ali (Member)

Date: / /2015

Signature:

Name: Prof. Dr. Abdul Monem S. Rahma (Supervisor)

Date: / /2015

Approved for the Computer Sciences Department, University of Technology

Signature:

Name: Asst. Prof. Dr. Hala Bahjet (Head of the Department)

Date: / /2015

Supervisor Certification

I certify that the preparation of this thesis **“Analysis and Segmentation Digital Video to Detect Script Writing”**, was carried by *Teaba Wala Aldeen Khairi*, under my supervision at the university of technology/department of computer science in partial fulfillment of the requirement of the degree of MSc.in system software of computer science.

Signature:

Name: Prof. Dr. Abdul Monem Saleh Rahma.

Date: / /2015

Linguistic certification

This is to certify that this thesis entitled **"Analysis and Segmentation Digital Video to Detect Script Writings"** was carried out by **"Teaba Wala Aldeen"** under my linguistic supervision. It was amended to meet the style of English language.

Signature:

Name: Dr. Ahmad AL-Beiruti.(Asst. Professor)

Date: / / 2015

Abstract

Traffic Road-Signs contain useful information for roads users, the operation of many modern applications like the automatic or smart vehicles require an automatic discrimination of the texts of the traffic road-sign. Discrimination of text is composed of several stages, the first of these stages is detection and extraction of the texts. In this thesis, an algorithm is developed to detect, locate, and segment of the texts and the words in the video clips. Specifically, this thesis is concerned with the texts that exist in the road signs in the city of Baghdad. These road-signs are analyzed and their features are determined.

In this thesis, a software program has been developed, the task of this program is to detect, localize and extract the texts of the road signs which may appear in a video clip. The proposed program is developed as an m-file by using matlab/2014b under Microsoft Windows 7- 64bit Professional operating system. With a laptop of core i7 processor and 8 GB RAM.

The sample subject was taken from a mobile camera in a moving car in different weather, day time. The data set contain different streets and a combination of road-signs.

The proposed approach includes two stages, the first one is processing the image to locate and extract images of the road-signs and neglect the rest of the image, and the second stage is processing the image of the road-

sign plate to detect and extract the texts without symbols and shapes. The advantage of this arrangement is to reduce the execution time of the algorithm. The basic structure of the algorithm depends on the following functions: edge-detector, dilation, filling-hole, and morphological-opening.

In order to evaluate performance and measure robustness of the proposed algorithm, it is subjected to many tests, firstly it was tested based on frames of the videos, in most of the tests(ten samples) the **recall-rate(r)** is equal or closed to**100%**with a total recall values of **89%**,also **precision-rate(p)**.In most cases equals or about **100%** with total **precision** value of **93%.**The algorithm is, then, tested based on video clips, implementation of the algorithm based on video clips confirms its ability to detect the texts which may appear in the video scenes, **recall-rate(r)** is excellent with a total value of **94.5%** and a total **precision** vale of **86.5%.**

The process time for each frame varying between (**1.5sec.-2.2sec**.) it depends on the complexity of the video scene.

The algorithm is tested to measure its validity in order to work under real-time operation, and by processing one frame and then exceed a set of next frames, this test shows that the algorithm is able to work under real-time operation with a total **recall-rate(r)** of **93%**.

List of Contents

Chapter 3: The Proposed Algorithm for Text Detection and Extraction

Chapter 4: Experimental Results and Implementation Analysis

Chapter 5: Conclusions and Suggestion for Future Works

List of Figure

List of Algorithm

List of abbreviation

Abbreviate	Meaning
CC	Connected Component
CED	Canny Edge Detection
CV	Computer Vision
DSSs	Digital Still Store Devices
DV	Digital Video
DVEs	Digital Video Effects
ED	Edge Detection
GB	Giga Byte
HMM	Hidden Markov Model
Matlab	Matrix Laboratory
MSER	Maximally Stable External Regions
OCR	Optical Character Recognition
RAM	Read Only Memory
RGB	Color Model
RS	Road Sign
RSs	Road Signs
SE	Structure Element
SVM	Small Vector Machine
SWT	Stroke Width Transform
WMS	Web Map Service

Chapter 1

Introduction and Historical Review

Chapter 1

Introduction and Historical Review

1.1 Introduction

Development in the field of adapting electronic devices to amend and suggest alternatives to be adopted as a substitute for human element in making decisions. There were several methods to transfer the information to a computer (the alternative to human element) some of which are sensors and cameras. Being directed to a computer system work inside the car to be a guide for special signs hints that distinguish roads specially road signs hints, that motivate an attitude for the researcher to distinguish road signs through photos of portable cameras that are to be set later on in the car.

Text detection is defined as the task that localizes text in complex background without recognize the characters individually [1]. Text plays a dominate role in video understanding as a text caries valuable important information related to the video contents. Text is the first object in video humans will pay attention to as a studies has been shown. Text detection pertain to the segmentation of text region in a video frame by identifying exact boundaries of text lines. Text detection plays a fundamental role in subsequent steps, such as, text localization, text word segmentation, and text recognition. A large size of data is existed in images, that's why the accurate detecting and identifying the text region is very important before performing any character recognition. The large changeability of the appearance of the text like (a complex background, different font styles, and variable size),make the text extraction from images a very rough duty [2,3].

The procedure of detecting text in image and video can be important step to a systems use the output text to complete their tasks, such systems are Optical Character Recognition(OCR) system, plate number identification number, document writing recognition, driver system assistant, and other[4].

1.2 Literature Review and Related Work

A review of the previous work in text detection for both image and video, detection and recognition of road sign text and natural scene text will be given below with a brief description to each of them:

In 2004, Wen W., Xilin Ch. and Jie Y. proposed an algorithm in order to detect text as in the proposed approach:

1- Selecting features by using some spatial criteria, these features will be used in some steps and some of will be put to track.
2- Using two criteria to detect possible candidates of road-signs panel from the clues of features found in the current frame and tracked features from previous frames. The two criteria are color information and vertical plane property of RSs.
3- Building pyramidal multi-scale images: The constructed multi-scale images of the current frame will serve as input for the following incremental text detection.
4- Incrementally detect text: Both new candidates and previously detected sign planes will be examined by a method of edge-based text detection on the pyramidal multi-resolution image(s). The Detection results from dissimilar

levels will be combined to detected text regions of original scale. The detection results can be obtained by merging matched text regions.

5- The detected text regions and features there will be tracked.

The approach efficiently incorporates tracking and detectioni mechanisms into the same framework. The images used in the experiments consist of most highway images only. In some situations sign detection is relatively more difficult as the signs are not clearly visible due to various reasons. The vertical plane did not be detect and the twisted sign planes also [5].

In 2005, Wen W., Xilin Ch. and Jie Y. proposed an algorithm in order to detect text in road signs the proposed work has two main steps as given below:

1- First step: applying a Divide-and-Conquer this strategy has been used to decompose the original task to two subtasks that localize of the road-signs and to detect the text. The two sub-tasks algorithm are smoothly combined into a joined framework through a real-time tracking algorithm.
2- Second step the framework provides text-detection method from video by mixing 2D features in each video frame (e.g., color, edges, texture) with 3D information obtainable in the video sequence (e.g., object structure).

This framework gives an overall text detection rate of 88.9% and a false hit rate of 9.2%. This frame work did not implement under real time condition. The shapes of road-signs was limited[6].

In 2006, A. VÁZQUEZ, R. J. LÓPEZ, S. LAFUENTE, P.GIL proposed an Algorithm in order to detect text in road signs the proposed work has three main steps as given below :

1- Basic color segmentation and shape classification to detect possible rectangular planes.
2- Every detected plane is extracted from the original image and then reoriented.
3- Chrominance and luminance histogram analysis and adaptive segmentation is carried out, and connected components labeling and position clustering is finally done for the arrangement of the different characters on the panel.

Experimental results shown that these steps strongly depends on correct separation between the background and foreground objects of the panel. The algorithm works with only white or blue panels[7].

In 2011, Gonz´alez A., Bergasa L.M, Javier J. and Almaz´an J. Presented a method in order to extract the text existed in road panels in street images .The steps of the proposed work are listed below:

1- Detect text on the image: uses MSER to detect characters and then traffic panels.
2- Character and word recognition: this step based on computing histogram of gradient direction over the edge points of the binaries objects.
3- The word recognizer: an HMM approach for computing the mostly probable model that has produced the set of observations for a further recognition. In order to growth the effectiveness of the recognition algorithm, the use of a Web Map Service (WMS) to resize the dictionary into smaller size.

A draw back of the proposed work is Algorithm works with just Blue color panel, Numbers and symbols detection needs to be improved, as well as character recognition at long distance. This frame work has a limited distance detection rate it did not work properly with long distance panels. In short distance the text detection rate 92% [8].

In 2014, Jack G. , Majid M. Presented a method in order to extract the text existed in road panels in street images ,Scene structure is used to define search regions within the image to find the traffic road sign The steps of the proposed work are listed below:

1- Maximally stable extremal regions (MSERs) and hue, saturation, and value color thresholding are used to locate a large number of candidates, to be reduced by applying constraints based on temporal and structural information.
2- A recognition stage to recognize text within detected regions.
3- Individual text characters are detected as MSERs and are grouped into lines, before being interpreted using optical OCR.

The method is comparatively evaluated and achieves an overall recall 80% and precision 97%[9].

In 2014, Junhee Youn , Gi H., Kyusoo Chong Presented a method in order to extract the text existed in road panels in street images consist of these stages:

1- Image pre-processing and binarization.
2- Arrow regions are extracted by the proposed four-direction contiguous pixel measures, so called line scan method.
3- Corner points are detected by using a“good features to track”algorithm based on an extended Newton–Raphson method.

4- Least squares matching (LSM) algorithm is applied to the corner points to extract direction information.

In this paper the possibility for automatic direction information extraction from images was proved only under relatively ideal conditions [10].

The thesis presents a problem solutions the previous researchers did not cover. Such a problem none of the researcher above has spot a light on the detection of the road-signs and the text on the road-signs in the night with the darkness effects. Also, the problem of the incomplete road-signs that will be appear because of the road – sign will be out of the scope of the camera or have a suitable deformation. None of the researcher detect texts on the Iraqi road-signs plates and the both Arabic and English text.

1.3Aim of This thesis

The main objectives of this thesis can be summarized in the following steps:

1. Developing an algorithm that can be able to detect and a segment of English and Arabic scripts in natural scene videos. The detection includes numbers if it appears in the road-signs panel.
2. This study is concerned with texts of horizontal and vertical traffic road-signs panels in Baghdad city, taking into consideration the variables and the factors that can affect the performance of the algorithm. The Varity of road-signs include the different colors of it.
3. Measuring and evaluating text detection rate for the proposed algorithm in a variety of operating conditions. Such a conditions a cloudy weather or a dusty weather, a day time changing (morning, night, noon).
4. Investigating the validity of the algorithm in order to work under real-time condition.
5. Handle Incomplete Road-Signs panel’s problem in a short processing time.

1.4Thesis Organization

This thesis contains the following chapters in addition to this chapter (chapter one):
Chapter 2: “Basic Theory”. This chapter describes edge detection concepts, morphological concepts, segmentation and feature definition.

Chapter 3:” The Proposed Algorithm for Text Detection and Extraction”. This chapter, provides the algorithms used to design the proposed system.
Chapter 4: “Experimental Results”. This chapter gives results obtained from implementing the proposed technique on different image and video samples at different road sign shapes and different weather and lighting conditions.
Chapter 5:”Conclusions and Suggestion for Future Works”. This chapter introduces number of conclusions from this work and lists number of recommendations for future work.

Chapter 2

Basic Theory

Chapter 2

Basic Theory

2.1 Introduction

The need of Analysis a video scene provide a lot of important information that can be very useful. To extract these information there is a need to use methods that help to detect, locate and segment them to be analyzed and classified in further process. Through the past few years many techniques of detection and extraction text regions in video and image has been implemented [11].

Generally this chapter contain a description of the theory that has been used in the proposed thesis such as the canny edge detection and the morphological operation and some tools to help in complete the output. A brief description on video processing and image processing also has been mentioned in this chapter.

2.2 Digital Video Files

Digital video technology become more current in the 1990s aura, in the 1970s, frames of analogue video were converted into digital form and altered for special effects [12].

Digital Video (DV) denotes to the process of the (capturing, manipulation, and storage) of motion images that can be displayed on computers screen [13]. Image in still form is a spatial distribution of intensity (Intensity can be defined as a measure over some interval of the electromagnetic spectrum of the flow of power that is radiated from, or incident on, a surface. It is measured in watts per square meter [14]) that is constant with respect to time [15]. Video, on the other hand, is a special intensity pattern that changes with time. Another common term for video is image sequence, since video can be represented by a time sequence of still images .Video has traditionally been captured, put in storage, and transmitted in analog form at the beginning [16,17].

2.3 Image File Formats

Images are organized on particular standards called image file format standards [18]. Binary images are encoded as a 2-D array, typically using 1 bit per pixel, where a 0 usually means “black” and a 1 means “white”. The main advantage of this representation is the image has a small size. To convert a gray image to binary image find the threshold value, If the value at the pixel position is greater than the threshold value then the value will be 1(white) else zero (black)[11].

Gray-level images are also encoded as a 2D array of pixels, usually with 8 bits per pixel, where a value of 0 corresponds to “black,” and 255 to “white,” and intermediate values indicate varying shades of gray[17]. The process of converts RGB values to grayscale values by forming a weighted sum of the R, G, and B components using this formula [19]:

$$\boldsymbol{New\ image} = (\mathbf{0.298} \times \boldsymbol{R}) + (\mathbf{0.587} \times \boldsymbol{G}) + (\mathbf{0.114} \times \boldsymbol{B}) \qquad \textbf{... (2.1)}$$

Where R refers to read, G refers to green, B refers to blue.

Colored images are more different and complex, the common method to store a color image contents are RGB representation. Each pixel in RGB representation is represented by a 24 bit number containing the amount of its red (R), green (G), and blue (B) components. [17, 19].

2.4 Image Segmentation

Segmentation can be described as the breakup of an image into a set of meaningful homogeneous region(s) the pixels of each region has the same properties, such as (gray-levels, contrasts, spectral-value, and textures properties). The outcome of segmentation is a specific homogeneous-regions with a unique label [20, 21].

The objective of this operation is to partition an image into separated regions which ideally correspond to interested objects (e.g., video texts) and accordingly obtain a more meaningful representation of the image [2, 22]. Segmentation is one of the important CV processes that is used in many practical applications such as machine vision, object recognition, surveillance, content-based browsing and else [16].

2.5 Feature Extraction

Extracting features is an imperative way in image analysis [23]. When the data need to be tested the amount of information is huge, and a small part of it contain the information that importance [24]. Feature extraction is a procedure to solve this problem by converting the data into another, reduced, representation set of features. The features to be extracted are chosen in such a way such that the set of features still contain information necessary to solve the task at hand [23, 22].

2.6 Edge Detection of an Image and Canny Method

Edges and contours plays a huge role in human vision systems. Edge is often describe or reconstruct a complete figure from a few key lines [25]. The Applying of the edge detection can be summarized by these reasons [26]:

- Identifies-discontinuities or sharp boundaries within an image which can help in later interpretation of the image.
- Reduces the volume of data making subsequent processing simpler.

In image analysis ED is most commonly used operations, the outline of an object can be formed by ED. The subject that interests is called object(s) in image analysis and vision-systems. If the edges in any image can be outline in accurate way the objects can be located easily, and some basic properties of these object such as the area, the perimeter, and the shape can be calculated [26].

The main goal of ED algorithms is to find the most relevant edges in an image or scene. These edges should then be connected into meaningful lines and boundaries, resulting in a segmented image containing two or more regions. Machine vision system will use the segmented results for tasks such as object feature extraction, and classification [17]. Edge detection is part of a process called segmentation the identification of region(s) within an image [26]. ED methods relying on the calculations of the first or second derivative along the intensity profile or the calculation of gradients [17].many edge detectors has been used like sobel or prewitt detector but the best edge detector with the desired results was the canny edge detector. In 1986, John-Canny defined a set of goals for an edge detector and described an optimal method for achieving them. Edge detector must address three specified issues [19]:

- The error rate**:** The edge detector should respond only to edges, and should find all of them, no edges should be missed.
- The localization**:** The distance between the edge pixels as found by the edge. detector and the actual edge should be as small as possible.
- The response**:** There should be only one response to a single edge [25].

The main purpose of ED generally is to reduce the unnecessary amount of image data with maintain of the structural properties to be used in some further image processing technique. The Canny edge detector (CED) is one of the most famous, strong, and effective edge detection operators [24]. The canny algorithm can be described in **Algorithm (2-1).**

Algorithm(2-1) Canny Edge Detection
Input: image to be processed. **Output:** binary image with edges.
Step1: *Smooth the image:* Convolve an image g with a Gaussian of scale σ to blur and remove unwanted detail and noise σ is the standard deviation. Where $G(x,y) = \frac{1}{2\pi\sigma^2} e^{\left(-\frac{x^2+y^2}{2\sigma^2}\right)}$ **Step2:** *Find the edge strength*: Finding edge strength by taking of the gradient of the image. Gradients at each pixel in the smoothed image are determined by applying the Sobel-operator. **First step** is to approximate the gradient in the x- and y-direction respectively by applying the kernels g_x and g_y. **And next** finding the gradient angleThe gradient magnitudes can then be determined as an Euclidean distance measure by applying the law of Pythagoras : $$\lvert G \rvert = \sqrt{G_x^2 + G_y^2}$$ $$\lvert G \rvert = \lvert Gx \rvert + \lvert Gy \rvert$$ Where Gx= and Gy are the gradients in the x- and y-directions respectively. **Next step** the direction of the edges must be determined as : $$\theta = \tan^{-1}(\lvert Gy \rvert / \lvert Gx \rvert)$$

Step4: *Tracing the edge in the image using θ:* the next step is to relate the edge direction to a direction that can be traced in an image. Any edge direction calculated will be round up to the closest angle.

Step5: *Non-maximum suppression:* Trace the gradient in the edge direction and compare the value perpendicular to the gradient. Edge pixels stronger than the high threshold are marked as strong; edge pixels weaker than the low threshold are suppressed and edge pixels between the two thresholds are marked as weak.

Step6: *Hysteresis:* Applying of two thresholds a higher value T_2 and a lower value T_1 to obtain the final result. The fate of each pixel is then determined according to the following criteria:

- If $|G(x, y)| < T_1$, then the pixel is rejected and isn't edge pixel.
- If $|G(x, y)| > T_2$, then the pixel is accepted and it is an edge pixel.
- If $T_1 < |G(x, y)| < T_2$, the pixel is rejected except where a path consists of edge pixels that connects it to an un-conditional edge pixel with $|G(x, y)| > T_1$.

2.7 Morphological Image Processing

A techniques used to offer tools for describing, and analyzing shapes in images. It has been used for preprocessing or post processing operations both the images containing shapes of interest in addition to provide useful tools for extract image components [27]. Morph operator can involve a geometric analysis of shapes and textures in images. The basic operations in morphological operation are dilation and erosion, which can be mixed to make other operations, such as opening [26, 27].

2.7.1 Structuring-Element

The Structuring Element (SE) is represented as a small matrix, whose shape and size as it needed to apply a certain morphological operator to an image [21].

2.7.2 Morphological-Dilation

Dilation is a morphological operation whose effect is to grow or bolding objects in a binary image. The dilation of a set A by B, denoted A $\oplus$ B, is defined as [17]:

$A \oplus B = \{c|c = a + b, a \in A, b \in B\}$ … (2.2)

Where A represents the image being operated on, and B is a second set of pixels called a *structuring element*. The value of the output pixel is the maximum value of all the pixels in the input pixel's neighborhood. In a binary image, if any of the pixels.is set to the value 1, the output pixel is set to 1 [17, 27].**Algorithm (2-2)** describes the dilation operation.

Algorithm(2-2) Dilation
Input: gray image A, structuring element B. **Output:** Image $A_2 = A \oplus B$.
Step 1: Read image(A); Convert A to binary; **Step2**://initialize A_2 same size as A// For ii=1:aa For jj=1:bb A_2(ii,jj)=0; End jj; End ii;

Step3: Apply SE on pixel(II,JJ) of the matrix A // If the origin of the structuring element coincides with a 'white' pixel in the image, there is no change; move to the next pixel. If the origin of the structuring element coincides with a 'black' in the image, make black all pixels from the image covered by the structuring element //

```
For II = 1 to (aa-1)          //aa=row  size A & bb=column  size A//
    For JJ = 1 to (bb-1)
        If(A(II,JJ)==1)
        For ii1=0:1
            For jj1=0:1
            If(A2(ii+ii1,jj+jj1)≅1)
            A2(ii+ii1,jj+jj1)=B(ii1+1,jj1+1);
          End ii1;
       End jj1;
   End JJ;
End II;
```

Step5: Display A_2; //Dilated binary image//

2.7.3 Morphological Erosion

Erosion operation cause a thinning to the objects in a binary image. The erosion of a set A by B, denoted as A ⊖ B, is defined as [17]:

$$A \ominus B = \{c \mid (B)_c \subseteq A\} \quad \text{...... (2.3)}$$

The value of the output pixel is the minimum value of all the pixels in the input pixel's neighborhood. In a binary image, if any of the pixels is set to 0, the output pixel is set to 0[24], **Algorithm (2-3)** describes the morphological Erosion.

Algorithm(2-3) Erosion

Input: Gray image A, Structuring element B.

Output: Image $A_2 = A \ominus B$.

Step1: Read image A;

Step2: Convert A to binary;

Step3: Initialize A_2,A_3,A_4;

Step4: Apply SE on every pixel(ii,jj) on image A//If the origin of the structuring element coincides with a 'white' pixel in the image, there is no change; move to the next pixel. If the origin of the structuring element coincides with a 'black' pixel in the image, and at least one of the 'black' pixels in the structuring element falls over a white pixel in the image, then change the 'black' pixel in the image (corresponding to the position on which the center of the structuring element falls) from ‘black’ to a 'white'. //

```
For ii=1 : row-1
    For jj= 1 : column-1
        A3 = [A(ii,jj)*B(1,1)A(ii,jj+1)*B(1,2);A(ii+1,jj)*B(2,1)
        A(ii+1,jj+1)*B(2,2)];
        A4(ii,jj)=min(min(A3));
        A2(ii,jj)=max(max(A3));
    End jj;
End jj;
```

Step5: Display A_2;

2.7.4 Effects and Uses of Erosion and Dilation

It is clear that erosion reduces the size of a binary object, when the dilation increases it. Erosion has the effect of removing small isolated features and reducing the size of objects by 'eroding' them at the boundaries [24]. One of the simplest applications of dilation is to join together broken lines which form a contour delineating a region of interest. For example, the effects of noise, uneven illumination and other uncontrollable effects often limit the effectiveness of basic segmentation techniques like edge detection technique [27].

2.7.5 Morphological Opening

The morphological opening of set A by B, represented as A ∘ B, is the erosion of A by B followed by the dilation of the result by B [17].

$$\mathbf{A \circ B = (A \ominus B) \oplus B} \qquad \text{……..… (2.4)}$$

It is used to remove thin extensions from objects or clean unwanted pixels (smoothing the contour) not causing any shrinking to the objects [17], **Algorithm (2-4)** describes the Opining operation algorithm.

Algorithm(2-4) Opining
Input: Image A, structuring element B. **Output:** Image $A_2 = (A \ominus B) \oplus B$.
Step1 : apply $A_1 = A \ominus B$; **Step2:** apply $A_2 = A_1 \oplus B$; **Step3:** Display A_2;

2.7.6 Morphological Region Filling

The binary image usually is an output of a previous threshold operation, in image processing application. This procedure is not perfect after these operations, and may leave holes. A *hole* is a background region surrounded by a connected Border of foreground elements [28]. The operation begins with assigning the value 1 to a pixel p inside the object boundary and start to get bigger and growing by performing the iterative dilations under a limited condition restricted by A^c. If the restriction is not placed, the growing process would flood the entire image area [29]. in an object region assuming that region are surrounded by an 8-connected boundary A, Let p be a pixel in it. The region filling goal is to fill up the entire region with 1's using a point, to start with, P [30]. morphological Region filling can be mathematically expressed as follows:

$$\mathbf{X_{kk} = (X_{kk-1} \oplus B) \cap A^c} \qquad \textbf{……... (2.5)}$$

Where kk = 1, 2, 3 … and X_0 = p as starting value, and B is the Cross Shaped SE. The algorithm stops at the kk^{th} iteration if $X_{kk} = X_{kk-1}$, union operation of X_{kk} and A contains the original boundary A and all the pixels within it labeled as 1. The algorithm works by identifying a point (X_0) within the hole and then increasing this region by successive dilations. The intersection (logical AND) is taken after each dilation with the logical complement of the object A is. With this step the filled region would simply grow and conquer the entire image. The intersection with the complement of A only allows the object to grow within the confines of the internal boundary. When the region has grown to the extent that it touches the boundary at all points, the next iteration will grow the region into the boundary itself. The intersection with A will produce the same pixel set as the previous iteration, at which point the algorithm stops, the structuring element must not be larger than the boundary thickness. Final step the union of A with the grown region gives the entire filled object A ∪ X_{kk} [19, 17]. **Algorithm (2-5)** illustrates Region Filling steps.

Algorithm(2-5) Reign filling
Input A= binary image, B= cross shape SE. **Output** A_2 a binary image with filled holes.
Sep1: Form an array X_0 of 0s, except the locations in X_0 corresponding5to the given5point in each hole, which is set to 1. **Step 2:** While X_{kk-1} <>X_{kk} Do For I=1 to X_{kk-1}; Add all neighboring pixel to X_{kk}; Then Remove every boundary A from X_{kk}; End For; End While. **Step3:** A= X_{kk-1} ∪ A

2.8 Extraction and Labeling of Connected Components

The word “extract” to an object in image, it means that the pixels that make it up has been identified. The process is done by generating an array with equal size as the image originally input and assign a label to each pixel. The pixels that belongs to the object are numbered the same label number and all pixels that belong to the background are numbered as a different label number. Another symbols can be used for labeling like letters or colors but it is only has symbolic meaning only [28].

Figure (2.1) showed an example of labeled image. Morphological operation has been used to extracting and labeling the connected components (in binary images objects. Extract a set of pixels in an image that all the pixels of a component are spatially connected with in the image [29]. All pixels with value 1 (foreground pixel) will be extracted in a group based on the connectivity. **Figure (2.2)** show an example of labeled component [19].

0	1	1	0	1	1	1	0
0	1	1	0	1	1	1	0
0	1	1	0	1	1	1	0
0	0	0	0	0	0	0	0
0	0	0	0	1	1	1	0
0	0	0	0	1	1	1	0
0	0	0	0	1	1	1	0

Figure 2.1 Image with 3-connected Components.

0	1	1	0	2	2	2	0
0	1	1	0	2	2	2	0
0	1	1	0	2	2	2	0
0	0	0	0	0	0	0	0
0	0	0	0	3	3	3	0
0	0	0	0	3	3	3	0
0	0	0	0	3	3	3	0

Figure 2.2 Labeled Connected Components.

Let A be a set one connected component or more than one, and p ($p \in A$) be a pixel to start with. To find all pixels in object component can be achieved using an iterative procedure, mathematically can be expressed as [30]:

$$X_{kk} = (X_{kk-1} \oplus B) \cap A \qquad \text{……………} \ (2.6)$$

kk = 1, 2, 3,…

Where $X_0 = p$ and B is a proper SE, The algorithm stops at the kk^{th} iteration if $X_{kk} = .X_{kk-1}$ [30]. **Algorithm (2-6)** shows the steps of labeling the connected component in an image.

Algorithm(2-6) Object labeling
Input Image A **Output** Labeled Matrix and Object Matrix Can be Extracted
Step1: Step1.1: Read A. **Step1.2:** Convert A to binary image. **Step2:** Define SE= B. //B a Structure Element// **Step3:** Initialize L with a zero value. // Label matrix Full with zeros//**.** **Step4:** In A Find a non-zero element position. **Step5:** create a matrix A2 with component of zeros and put value 1 in the element position with non-zero value has been found in **Step 4**. **Step6:** Apply dilation using B on matrix A_2. //see algorithm (2-4)// **step7:** //Apply intersection with A// New A= A& dilate (X, B). **Step8:** Check whether New A==A_2. If no, then A_2=New A and Apply steps 6 and 7.

Else stop the iteration.

Step9: In New A Find the non-zero elements position.

In matrix L place a number N in those positions.

Step10: As same as **Step9** place zero in the same positions in the input matrix A

Step11: Find a non-zero element position in the matrix A.

If found, **go to step 5**

Else hold the iteration.

Step12: The connected components can be extracted by using the Labels.

2.9 Properties of Object Region

A region of interest is a connected set of pixels carrying the value 1.the connected property means that form any 1 pixel, there is a path of 1's to any other 1 pixel in its region of interest. Regions of images (also called objects or connected components) can be contiguous or discontinuous. These regions can have properties, such as an area, orientation, center of mass, and bounding box [19].

1. Region Area: the region pixels number is represent area of the object[19].area can be defined as :

$$\boldsymbol{Ar} = \sum_{i=1}^{N} \sum_{i=1}^{M} \boldsymbol{B}[\boldsymbol{i}, \boldsymbol{j}] \qquad \textbf{....................(2.7)}$$

2. Region Bounding-Box: the smallest rectangle containing the region [31].

3. Region Extent: the ratio of pixels in the region to pixels in the total bounding box [31]. Computed as the Area divided by the area of the bounding box. This property is supported only for 2-D input label matrices [19, 17]. Algorithm (2-7) shows the region property area, bounding box, extent algorithm.

Algorithm (2-7) Region Property

Input binary image.

Output Area, Bounding box, Extent.

Step1: //Region Area//

Step1.1: Read image im(I,J).

For Ii= 0 to width

For Jj = 0 to height

If im(Ii,Jj)== 1 then

A=A+1;

End If.

End For.

End For.

Step1.2: Return A.

Step2: //Region Bounding Box//

Find the row and column having pixel value one [row,colun]=find(label==1)

Step2.1: Assign start position(R_x,C_y) by:

R_x = Minimum value of row - 0.5.
C_y = Minimum value of column - 0.5.

Step2.2: Find width
W= (Maxvalue of column-Minvalue of column+1)

Step2.3: Find box Length

L= (Maxvalue of row-Minvalue of row+1).

Step2.4: Bounding-Box=[R_x,C_y,W,L].

Step3://Region Extent// Ext= area/bounding box area.

2.10 Image Crop

The operation of cutting a pre-defined part of the image is cropping the image (in different definition cutting out a small rectangular of the image). This may involve cutting some of the image from the left, right, top, bottom, or any mixture [19]. **Algorithm (2-8)** describes image crop steps.

Algorithm (2-8) Image Cropping Algorithm
Input Image **Output** Cropped Portion of an Image
Step1: Read Image im. **Step2:** Input upper left corner b(X_b,Y_b). **Step3:** Input width and high of new image b(W_b,H_b). For ii= X_b to (X_b+H_b) For jj=Y_b to (Y_b+W_b) b(ii,jj)=im(X_b+ii,Y_b+jj); End For. End For. **Step4:** Display the cropped image.

2.11 Terminology of the Evaluation

In order to evaluate the performance and measure the robustness of the present algorithm, the following terms and ratios are used[9]:

1- **Recall(r):** this rate is calculated by dividing the correctly detected texts to the sum of false-negative and correctly detected texts. The term false-negative (or missed text) means the region which is indeed a text, but the algorithm has not detected it.

$$r = \frac{C_t}{C_t + M_t} \times 100\% \qquad \text{……...... (2.2)}$$

Where:

C_t = Correct Detected Text.

M_t= Missed Text(false-negative).

2- **Precisions (p):** this is calculated by dividing the correctly detected texts to the sum of false-positive and correctly detected texts. The term false-positive means the region which is indeed not a text, but the algorithm has detected it as a text.

$$p = \frac{C_t}{C_t + F_t} \times 100\% \qquad \text{………..… (2.3)}$$

F_t= False Positive.

3- **Detection rate (DR)**: is defined as a ratio of the detected texts to the total texts in the image detection rate equals to the recall rate.

DR= r. **…………… (2.4)**

4- **Error rate(ER):** is defined as a ratio of un-detected texts (false negative) to the total texts in the image.

ER=100-DR. **…………… (2.5)**

5- F-Measure(F_s): is the weighted average of the recall rate and precision rate, the value of F-measure is confined between 0 (worst value)and 1(better value).

$$F_s = 2 \times \frac{r \times p}{r + p} \quad \text{……….. (2.6)}$$

Chapter 3

The Proposed Algorithm for Text Detection and Extraction

Chapter 3

The Proposed Algorithm for Text Detection and Extraction

3.1 Introduction

Detect and extract text from natural scene video is a very hard and complex process in most of the natural scene videos, difficulty of this issue comes from the need to process a stream of enormous amount of data. Applying of detection and extraction techniques on detect texts of the road-sign adds other problems to the subject where many factors will affect the possibility of detection and locating the text, some of these factors are:

1- Vibration of the moving camera.

2- Diversity of road-signs types.

3- Complexity of background of the scene (trees, dirt on the road sign).

4- Lighting level and the weather conditions.

In this chapter, the proposed text detection method will be introduced with an explanation of the algorithms. **Figure 3.1** shows a general graphical chart of the algorithm. The thresholds are explained in detail in appendix A.

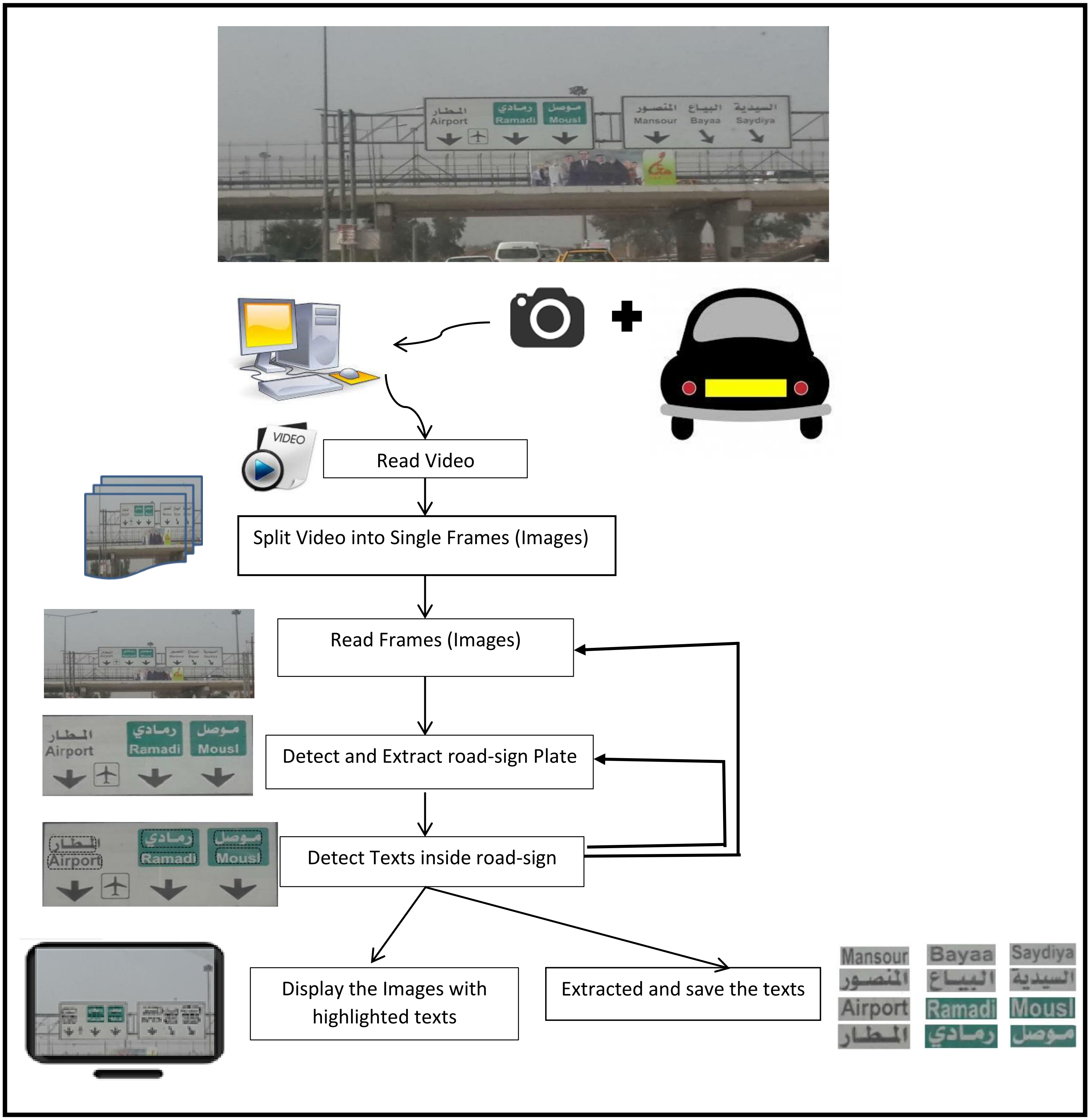


Figure 3.1 Graphical chart of the algorithm.

3.2 Road Sign Features

In order to identify the characteristics and features of road traffic signs, a general and compressive study has been made on the road signs in Baghdad city, the road signs can be categorized into two types:

- Side way road-sign: it is installed on poles beside the streets.
- Upper panel road-sign: it is installed on overhead metal bridge above the street.

Generally, road sign is of rectangular shape and it may be horizontal or vertical rectangle. **Figure (3.2)** shows types of road signs.

-a- Side way RS with blue background.

-b- Upper RS with white background.

-c- RS with background of more than one color.

Figure 3.2 Examples of different road-signs **a)** Side way RS with blue background. **b)** Upper RS with white background. **c)** RS with background of more than one color.

The dimensions of the road sign plate depends on the amount of traffic information that is displayed in it and some other factors. It has been observed that there is a variety of standard of the dimensions (height, width) of the road sign in Baghdad city, some of these standards are listed in **Table (3.1)**.

Table (3.1) dimensions of road signs

Height (cm)	Width (cm)	RS Image
200	450	
200 200	725 250	
300	300	
125	170	

Background color of the road-sign plate may be one of the following forms:

1. ***White***: This color is used as a background in the road-sign of the main and subset-streets.
2. ***Green***: this color is used for plates that refers to areas outside Baghdad.
3. ***Blue***: this color is for road signs of high ways lines.

It is possible to find a panel containing more than one background color, for example white and green as shown in **Figure (3.2c).** The information that may be appeared in road sign are:

1- Words in Arabic language, usually, placed in the top of the plate.

2- Words in English language, usually, placed under the Arabic words

3- Numbers.

4- Symbols such as denotation arrow and shapes indicate to airport or mosque, etc.

The font color of those items is white for plate of green or blue background while for white background the font color will be black **Figure (3-3)** shows a different RS plates. **Figure (3-3)** can present the levels of complexity that exist in the natural scene videos.

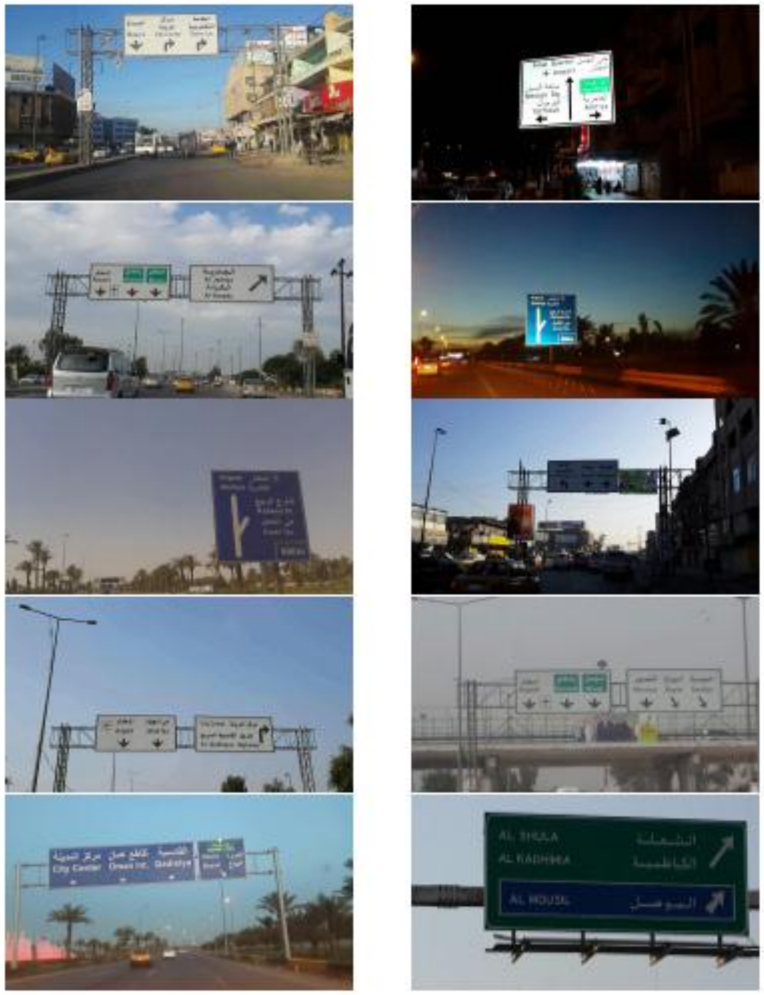


Figure 3.3 Road-Signs plates.

The plate of the road sign is surrounded by stripe frame, the color of this frame is black for white background and white for green and blue background. It is important to mention here, that the front face of the road-sign has intense ability to reflect the light .Therefor, in the dark, once a direct light is applied on the front face of the road sign, all the words and symbols can be seen clearly as shown in **Figure (3.4)**.

Figure 3.4 Road-sign at night.

3.3 General Layout of the Algorithm

In general the program consists of the following four main stages:

1- Preprocess.

2- Detection and locating the region of the road sign (or road signs).

3- Detection and locating the text (or texts) inside the road sign.

4- Post-processing (display the results).

Figure (3.1) shows a general graphical chart of the proposed algorithm.

Detail of these four stages is described in the following sections where explanation of the program and the algorithms will be supported with applied examples from a video sample. Figures will be show the execution of the successive steps of the program for this sample in this chapter. Source video which is taken as an applied example through this chapter (there are ten others in chapter four) has the specifications as illustrated in Table **(3.2)**. This example just to shows the results of the algorithm steps.

Table (3.2) Specifications of the source video

Resolution in bit per pixel	Video format	Video duration	Number of frame	Day time	Writing language	Weather condition	Car Speed
1920x1080	AVI (.avi).	3 sec.	72	noon	Arabic and English	dusty	60 K/h

3.4 Preprocess Stage

The videos which are processed in this study was captured by a digital camera that mounted on a moving vehicle. These videos saved on the hard disc of the computer. In this work, the program is adapt to be suitable for most of Video format like (AVI (*.avi), MPEG (*.MPG), MP4).The program can be adapted to process video of any resolution. The longest sample was with duration of 19 seconds. Preprocessing stage consists of the following sub-stages:

1- ***Converting the video into frames***: as illustrated in **Algorithm (3-1)**.

Algorithm (3-1) Reading and Converting the Video File
Input: Video File **Output:** Set of Images(Frames)
Step1: Start. **Step2: Read** source video. **Step3:** Convert video to single frames; **Step4:** Count the number of frames; Number of frames = Frame per second × Duration of video. **Step5: End**.

2- ***Pre-Process the frames***: the source video is framed into a number of frames, each frame of the source video is saved, with (Frame-x) label, in jpg format. This image is, then, read and converted into gray-scale. After that, the boundary of the objects in the image are obtained by using canny-function [**Algorithm (2.1)]**. Two thresholds have to be specified to adjust the operation of canny-function which are:

1) 2-elements sensitivity thresholds vector, those 2-elemnts are the low threshold ($T_{1=0.04}$) and the high threshold ($T_{2=0.10}$).
2) Sigma=3, which is the standard deviation of the Gaussian-Filter.

Figure (3.5) shows frame number-27 of the source video while **Figure (3.6)** and **Figure (3.7)** shows the gray-scale and canny-function of this image respectively. **See appendix A** table (1) for further information.

Figure 3.5 Frame number-**27** of the source video .

Figure 3.6 Gray-scale of frame number-**27.**

Figure 3.7 Canny edge detection of frame-27 .

The objects, appear in **Figure (3.7)** in term of its boundaries, thus, it is possible to extract the features of those objects like: shape, dimensions, area, etc. Also it is possible to note the presence of meaningless, small objects. To reduce the complexity of the image, the small unwanted objects is removed by using the morphological-opening function [see A**lgorithm (2.6)]**. Opening function cleans all connected components (objects) that have less than P pixels, so this functions has two inputs which are:

1) The binary image (to be cleaned).
2) The threshold $P_{1=130}$, which is a function of image resolution.

Figure (3.8) shows the image after applying morphological-opening function. **See Appendix A** table(1) for further information.

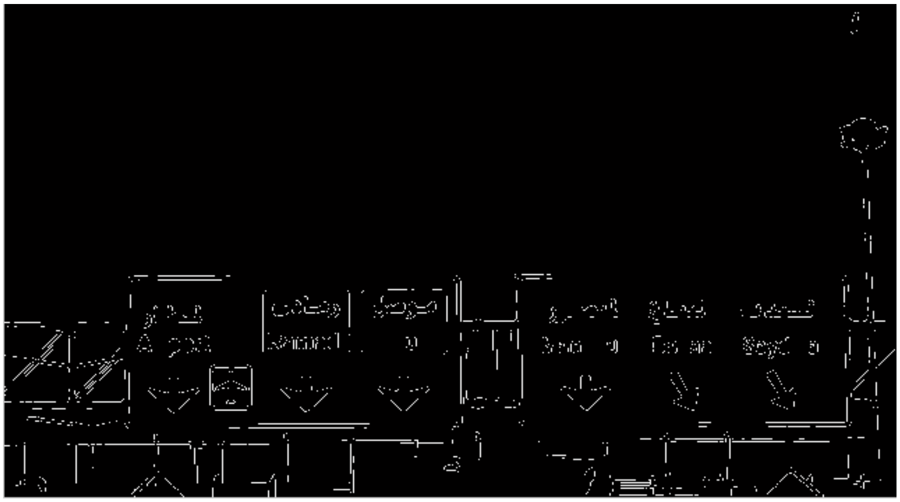


Figure 3.8 morphological opening of frame-27.

All steps of pre-process the frames are illustrated in **Algorithm (3.2). See Appendix A** Example 1 for further information on the remove the noise.

Algorithm (3-2) Preprocess the Frame
Input: Image (frame) of source video. **Output:** Edge-Detection of (IM) with applying Opening Function..

Step1: Start.
Step2: Save frame with (Frame-x) label and in jpg format.
Step3: Read image IM=frame[x].jpg.
Step4: Measure the size of image IM (height, width).
Step5: Set MS=zero matrix (same dimension of IM (h, w)).// MS is a gray-scale of IM//
    For Ih= 0 to high
      For Jw=0 to width
        Get Red(R), Blue (B), Green (G);

MS(Ih,Jw)= apply **equation(2.1);**

End Jw.

End Ih.

Step6: Find(g3)= binary-image with edge by applying Cnny Edge Detection on image (MS) according to T_1, T_2 and sigma thresholds.

Step7: Remove all connected objects of less than P_1 pixels by using Morphological Opening-function of image (g3).

Step8:End.

3.5 Detection and Extraction of the Road-Sign Plate

To detect the RS there are several steps that will be explained in the next sections:

3.5.1 Detection of Complete RS Plate

As mentioned before the road signs have a rectangular shape with distinctive framework (border), therefore, the first step of this stage is to find the rectangular objects. Normally, after applying canny, the boundary of the road-sign(RS) appears as a complete and continuous rectangle. In some cases discontinuity may arise in the road-sign shape because of presence of defects in the RS itself, long shooting distance or another reasons which may deformation the image. It is important, here, to enhance the image to ensure appearing all the rectangles fully without any defects, to do this, morphological-dilation is applied.

The morphological-dilation [**Algorithm (2.2)]** uses a structure element (SE) object for expanding shapes contained in the input binary image. There are two characteristic that must be identified:

1) Shape of the SE.

2) Size of the SE.

To complete the gabs of the border for any rectangles, it is needed two dilation operations, the first dilation applied on the pair of vertical lines, line structure element is used with angle of 90°. The second dilation is applied on the pair of horizontal lines, also, line SE is used but with angle of 0° see **Figure (3-9).** The size of the two elements (length of line in this case) is chosen carefully to achieve the desired goal with minimum distortion of the rest parameters of the image. **Figure (3.9)** shows the image after applying the two morphological dilation.

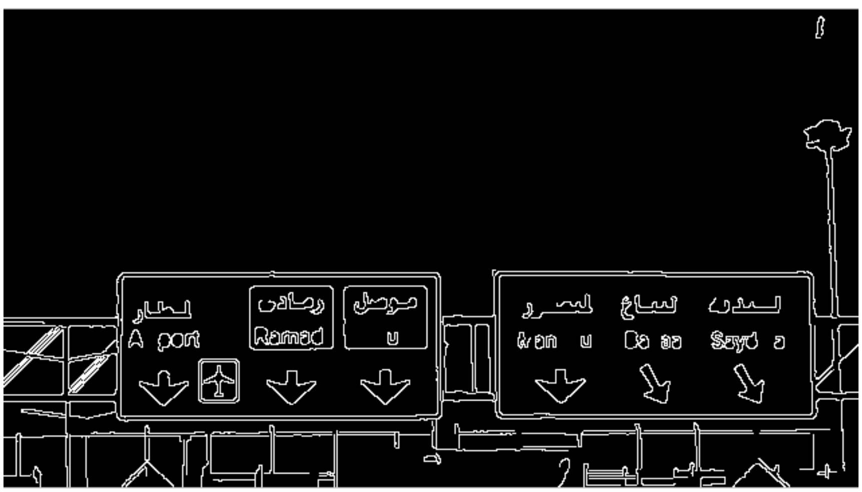


Figure 3.9 Dilated image.

The next step in this stage is to fill all holes that exist within the image, when a hole is a closed shape of background pixel which can’t be reached by filling from the outer edge of the image. As a result of this process, all closed area will turn into solid object as shown in **Figure (3.10)**. **See Appendix A** Example 2 for further information on the structure element effect.

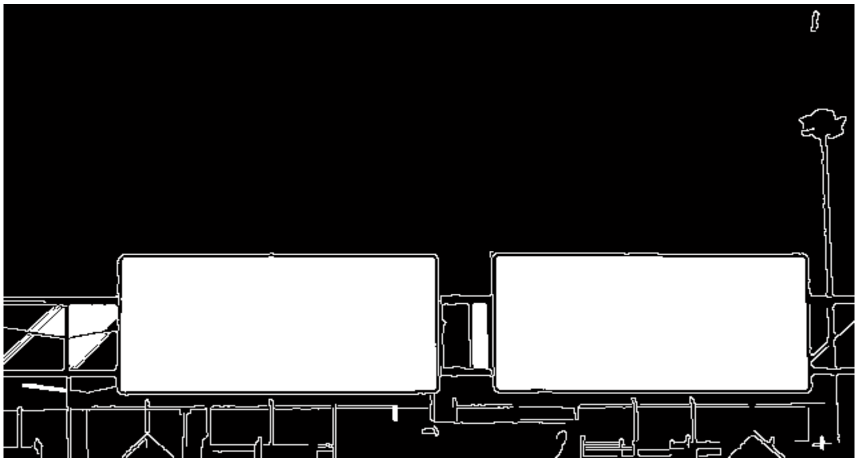


Figure 3.10 Image with filled holes.

It is obvious from **Figure (3.10)** that all the road sign areas are included in this action (that is required), in addition to some objects that don't mean anything. The objects that meaningless are removed after applying a conditional threshold. The strategy, that used to discriminate the road-sign are comprises:

1) Labeling the connected pixels objects [**Algorithm (2.6)]** where the background is labeled 0 and label 1 dedicated to the first object, and label 2 to the second object and so on.
2) Measuring properties of the objects [**Algorithm (2.7)]**, among many features provided by regionprops-function, it will be relied on only the required features to determine wither the object is RS or not. These features (Region-Property Features) and their descriptions are listed in **Table (3.3)**.

Table (3.3) required features to detect RS

Property	Description
Area	The number of pixels in the binary object
Bounding Box	The smallest rectangle containing the object
Upper Corner	Identifies the left upper corner of the boundingbox in form(x,y)
Dimensions	Identifies the dimensions of the boundingbox in form (horizontal-dimension, vertical-dimension)
Extent	Identifies the ratio of area of the object to area of the total bounding box

The object can be considered as a RS plate if the following conditions are true:

1. Extent value has to be close to unity which means the object is rectangle.
2. The area of the object has to be larger than a specific limit, this condition leads to exclude all small and meaningless objects also small and far RS plates that have tiny and unreadable texts. The threshold or the limit of the area is determined to be larger than 1 percent of the image area.
3. Ratio of object width to object height has to be approximately equal to one of the known ratios of plate width to plate height. This condition is based on the ratios not on the dimensions because the dimensions are change during the shooting distance while ratios remain constant. According to **Table (3.1),** ratio of road signs plate width to height may

be one of the following values: 2.25, 3.6, 1.25, 1, and 1.36. All the previous steps will appear in the main algorithm [see **Algorithm (3.7)]**. At the beginning the sub algorithm (**Algorithm (3-3), Algorithm (3-4), Algorithm (3-5), and Algorithm (3-6)**) must be read first to be clear and understand the main **Algorithm (3-7).**

3.5.2 Detection of Incomplete RS Plates

In the same video clips, it is possible to appear as an incomplete RS (reminder of the plate is outside the scope of photography), example of such cases is shown in **Figure (3.11).**

Figure 3.11 Incomplete RS plate.

Reviewing and analyzing of many video clips, give the following notes:

1. Some of the RS are incomplete in all frames of the video clip.
2. Some of RS are complete in a number of frames and incomplete in the others.

By applying the same procedure described in **Section (3.5.1**), incomplete plate will not be detected at all because for those RSs, and by applying the edge detector, only three sides of the framework will appear and the fourth side will be outside the scope of the image. Therefore, the solid object cannot be achieved, then the road sign cannot be detected. To detect incomplete RSs, an algorithm is proposed, the idea of this algorithm can be summarized as:

- **a.** Find the incomplete RS by seeking a pair of parallel lines which are perpendicular to one of the image edges.
- **b.** If the length of these two lines is taller than a specific value and the distance between them is more than a specific value, it could be considered that these two line are at opposite sides of the rectangle.
- **c.** To complete the fourth side of this rectangle, a line align to the image edge will be added between the two opposite sides**.**

Algorithm(3-3) illustrates the steps used to close the incomplete RS which had severed from the top of image. **Algorithm (3-4)** illustrates the steps used to close the incomplete RS which had severed from the left side of image. **Algorithm (3-5)** illustrates the steps used to close the incomplete RS which severed from the right side of the image. For the RS which had severed from the bottom of the image, another algorithm can be developed in the same style, but it was noted that such a RS will, certainly, appear as a complete RS in the next frames which means it can be detected as a complete road-sign in the next frames. Therefore, this work did not include an algorithm for signs which had severed from the bottom of the image.

Algorithm(3-3) Complete RS Which Severed from the Top of Image

Input: bw=binary image after morphological dilation, Image Width, and Image Height

Output: Binary image(bw) with closed rectangle from top of image.

Step1: Start.

Step2: Set Right Line=0;Flag=0;

For J=1 to ImageWidth-5

Step2.1: Evaluate SumTop=number of white pixel in the slice of bw(1:100,J:J+5);

Step2.2: IF SumTop>50 and J-RightLine>20 **Then** //distance between 2 rectangles have to be at least20pixels//

Flag=Flag+1;//there is a nominee line there//

Step2.3: IF Flag =1 **Then**

LeftLine=J;//specifying location of left side of the rectangle//

Step2.4: IF Flag=2 and J-LeftLine>25 **Then** //distance between 2 sides have to be at least 25 pixels//

RightLine=J;//specify position of right side of the rectangle//

bw(1:5,LeftLine-5,RightLine+5)=1;//create a line, align to the top edge of the image, between the two sides of the rectangle, with thickness of 5 pixel width//

Flag=0;//reset flag value for next rectangle//

End IF.

End For.

Algorithm(3-4) Complete RS Which Severed from the Left Side of Image

Input: bw =binary image after morphological dilation, image width, and image height

Output: Binary image (bw) with closed rectangle from left side of image.

Step1: Start.

Step2: Set LowerLine=0;Flag=0;

For I=1 to ImageHeight-5

Step2.1: Evaluate SumLeft=number of white pixels in the slice of bw(I:I+5,1:100);

Step2.2: IF SumLeft>50 and I>LowerLine+20 **Then**

Flag=Flag+1;//there is a nominee upper line here and this line have to be at distance of more than 20 pixels from image edge or lower side of neighboring rectangle//

Step2.3: IF Flag =1 **Then**

UpperLine=I;//specifying location of Upper side of the rectangle//

Step2.4: IF Flag=2 and I>UpperLine+25 **Then** //distance between the two side of the rectangle have to be more than 25 pixels//

LowerLine=I;//specify position of Lower side of the rectangle//

bw(UpperLine-5:LowerLine+5,1:5)=1;//create a line,align to the left edge of the image,between the two sides of the rectangle,with thikness of 5 pixels width//

Flag=0;//reset flag value for the next rectangle//

End IF.

End For.

Algorithm(3-5) Complete RS Which Severed from the Right Side of Image

Input: bw=binary image after morphological dilation, image width,and image height

Output: Binary image(bw) with closed rectangle from right side of image.

Step1: Start.

Step2: Set LowerLine=0;Flag=0;

For I=1 to Image Height-5

Step2.1: Evaluate SumRight = number of white pixel in the slice of bw(I:I+5,ImageWidth:ImageWidth-100);

Step2.2:IF SumRight>50 and I>LowerLine+20 **Then**

Flag=Flag+1;//there is a nominee of upper line here and this line have to be at distance of more than 20 pixels from image edge or lower side of neighboring rectangle//

Step2.3:IF Flag =1 **Then** UpperLine=I;//specifying location of Upper side of the rectangle//

Step2.4:IF Flag=2 and I>UpperLine+25 **Then** //distance between the two side of the rectangle have to be more than 25 pixels//

LowerLine=I;//specify position of Lower side of the rectangle//

bw(UpperLine-5:LowerLine+5,ImageWidth-5:ImageWidth)=1;

//create a line,align to the right edge of the image, between the two sides of the rectangle,with width of 5 pixels//

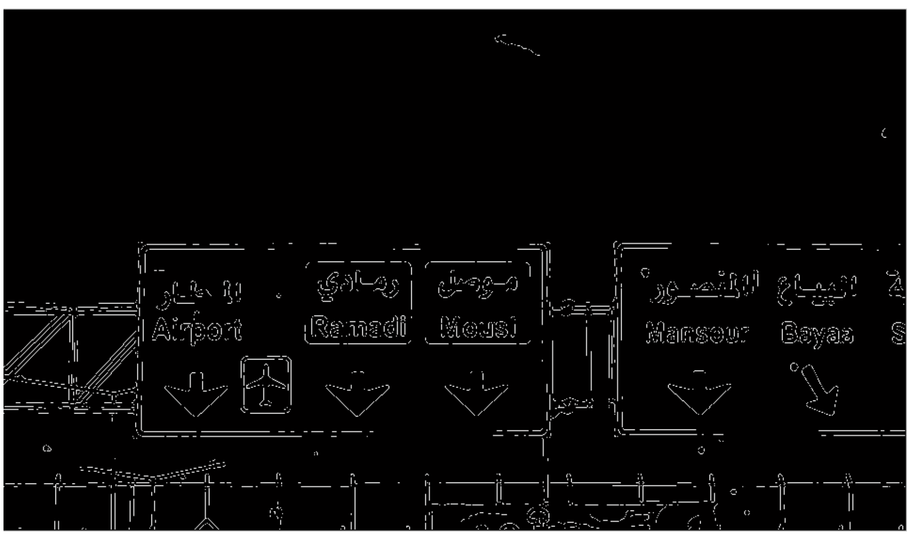
Flag=0;//reset flag value for the next rectangle//

End IF.

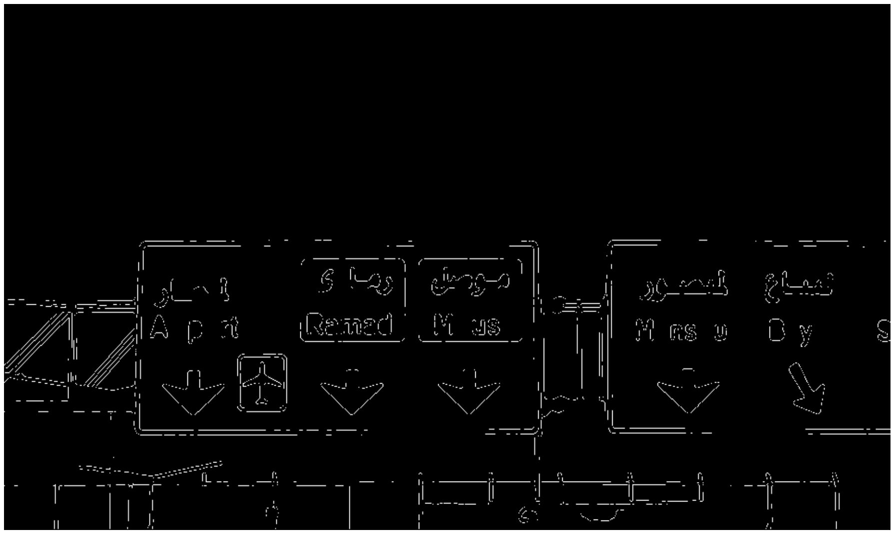
End For.

By completing the missing side of the incomplete RSs, these signs can be handled like any complete RS, and according to the steps outlined in **Section (3.5.1),** even the incomplete RSs will appear as a solid object. **Figure(3.12)** shows the steps of implementing the algorithm of incomplete RS.

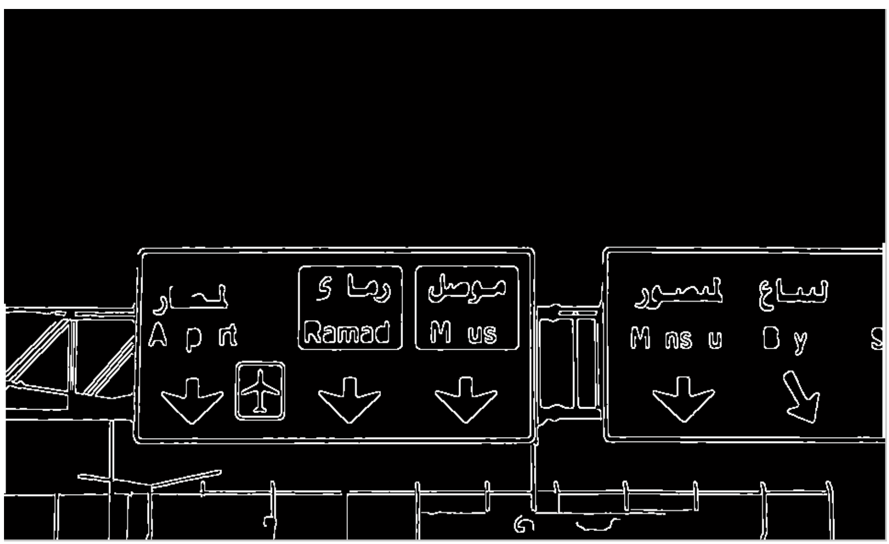
-a-



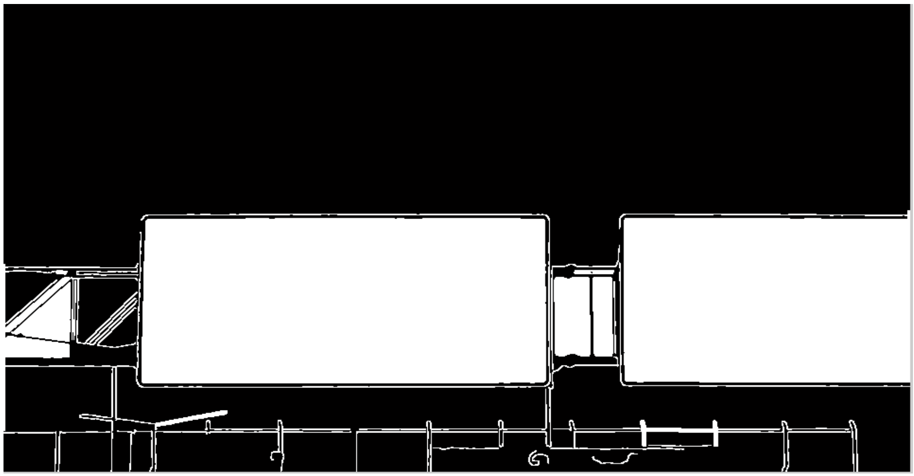
–b-



- c-



–d-



–e-

Figure 3.12 **a)**Incomplete RS, **b)**Canny-Edge Detection of incomplete RS, **c)**Clean the unwanted pixels, **d)**Dilated image with adding the missing side of the rectangle, **e)** Image with holes.

The conditions which determine whether the solid object is incomplete RS or not, are different from the conditions of the complete RS, because there are differences between the two cases in term of areas, dimensions and locations. The object can be considered as an incomplete RS plate if the following criteria's are true:

1. Extent value has to be close to unity (this conditions is common for the cases of complete and incomplete RS plate).
2. The solid object has to be tangent to one of the edges of the image.
3. Generally the area of incomplete RS is less than the area of complete RS. Neglecting the incomplete, the RS which has exposed portion of less than 30% of the total sign area. On this basis, the threshold or the limit of the area is determined to be more than 0.3 percent of the total area of the image.

3.6 Crop the Road-Sign Plates

The solid objects which satisfy the set of criteria mentioned in **Section (3.5.1)** and the objects which satisfy the set of criteria mentioned in **section (3.5.2)**, these objects, often represent RS plates. By using imcrop-function [**Algorithm (2.8)]** the RS are cropped, where this function is used to crop a portion of the original image (IM), this portion is appointed by the location and the area of the RS. The cropped portions will be saved as an image and passed to the next stage of the algorithm and the rest of the original image will be ignored. It is important to mention that in, this work, the proposed algorithm has the ability to crop all the RS plates in the image whatever the number. Examples of images with their cropped RSs are shown in **Figure (3.13).**

-a-

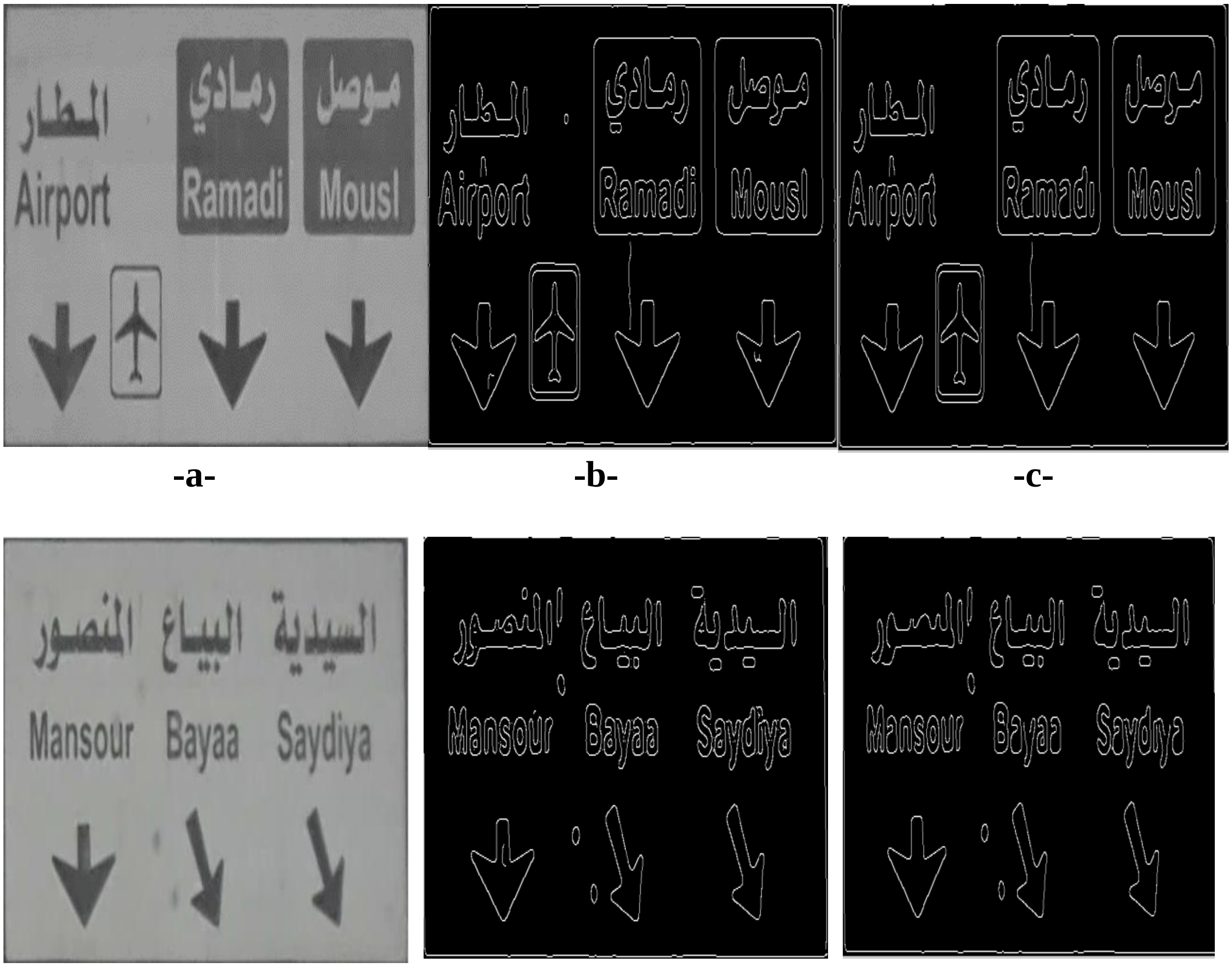
-b-

Figure 3.13 a) Image has 2-RS and the cropped plates, **b)** Image with complete and incomplete RSs and the cropped plates.

3.7 Detect and Segment the Texts inside the RS Plate

For each plate, which is detected and cropped by using the previous steps of the algorithm, the following procedure will be applied in order to detect the texts inside this RS plate:

1. The image is converted to grayscale binary image as shown in **Figure (3.14a**) and **Figure (3.14d)**.
2. The edge of the objects in the image are obtained by using CED as shown in **Figure (3.14b)** and **Figure (3.14e).**
3. Noise and un-wanted edges are removed by using the morphological open-function with threshold P_2 as shown in **Figure (3.14c)** and **Figure (3.14f)**.
4. Connecting all components and letters of the word to alter it to one, continuous and interrelated piece.



-a- -b- -c-

-d- -e- -f-

Figure 3.14 a**)** First Road-Sign plate. **b)** First Road-Sign plate after applying canny edge detector. **c)** First Road-Sign plate after removing small objects. **d)** Second Road-Sign plate. **e)** Second Road-Sign plate after applying canny edge detector**. f)** Second Road-Sign plate after removing small objects

English words consist of a number of isolated letters, so that dilation function is used with line SE at angle of 0̊, to connect letter with each other. The small letters (i) and (j) contain a point, again dilation function is used to connect the points with the rest of word, the SE, here, is line with 90̊ angle see **Figure (3.15)** for example.

Arabic words consist of one section, or two sections or even more, dilation function with line SE at angle 0 is used to link these sections as happened with English words. The letters(ت، ب، ذ ،ث،خ،ج، ز،ن،شن،ض،ظ،غ،ق،ي،ف) contain points and the letter(ك) contain Hamza " ء", as previously mentioned dilation function will connect the points with the rest of the word. In some cases, such as that letter (كـ) comes after the letter (ر) or after the letter (ز), the need is arisen to use dilation function with disc SE to achieve a connection between the letters. It is important to determine an appropriate values of the length of the line SE and the radius of the disc SE. For low values, the required connection between the components of one word will not be achieved. At high values, it is possible to overlap adjacent words with each other. These wrong and unsuitable dimensions of the SE leads to inaccurate detection. The dimensions of the SE have to be a function of word size, or in other words a function of the size (Area) of the RS. Studying and analysis of a variety of RSs and the texts inside them, the following empirical formulas are developed for the dimensions of the SEs:

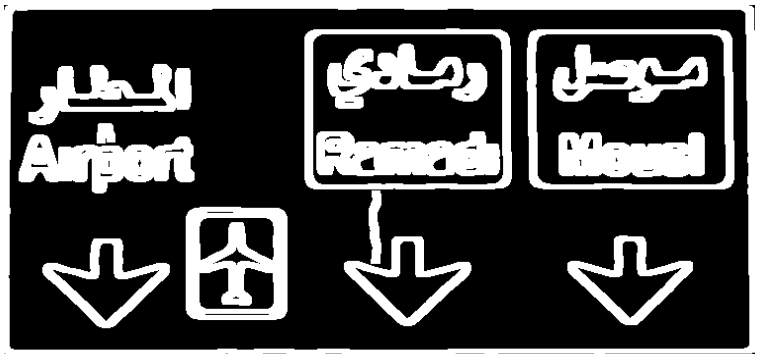
a. For line SE (at angle =0̊): Dimensions=$C_1 \times a \times resolution$
b. For line SE (at angle =90̊): Dimensions= $C_2 \times a \times resolution$
c. For Disc SE: Dimensions= $C_3 \times a \times resolution$

Where a=Area of the RS,

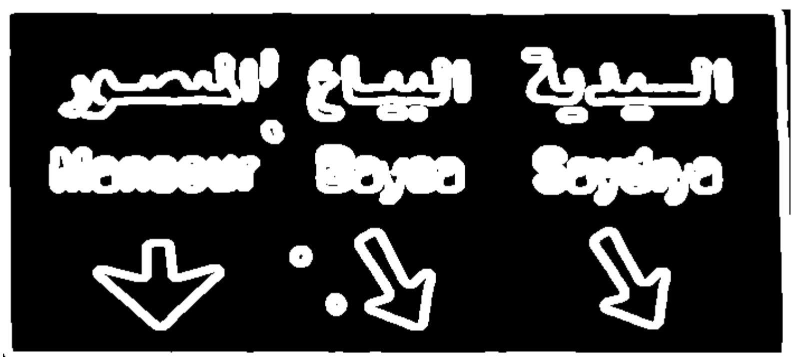
C_1, C2, C3 are constants,

Resolution= Resolution of the image.

The image after applying the dilation function is shown in **Figure (3.15)**. **See Appendix A** for further information.



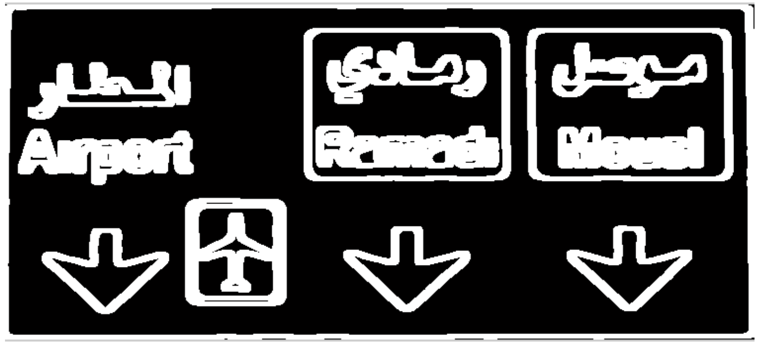
-a-



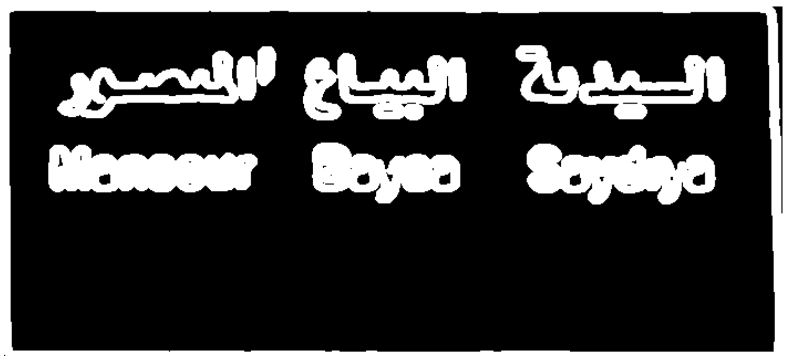
-b-

Figure 3.15 a) Dilated image for the first plate. **b)** Dilated image for the second plate.

Morphological opening function is applied, again here, to remove all the unconnected small object with threshold ($P_{3=1000}$). $P_{2=30}$ and $P_{3=1000}$ are a function of both of area (a) and the image resolution, but $P_3 >> P_2$. **Figure (3.16)** shows the image of the plates after applying morphological opening-function. **See Appendix A** for further information.



-a-



-b-

Figure 3.16 After applying morphological opening for **a)** The first road-sign plate. **b)** The second road-sign plate.

6. Labeling all the objects in the image by using bwlable-function.
7. Measuring the properties of all objects in the image by using regionprops-function see **Table (3-3)** gives an example of the values that has been used in the algorithm. These properties are: Area, Bounding-Box, Locations (upper corner of the bounding-box), and Dimensions (textheight, textwidth).

3.8 Post-processing: Extraction and Display the Results

Each object will be considered as a text if it satisfies the following conditions:

1- The area of the object has to be greater than 1.5 percent of plate area.

2- Object width has to be more than 1.2 times the object height. This conditions is based on the fact that all Arabic and English words are written horizontally.

By these two criteria's, the texts will be segregated from the rest symbols and signs in the RS plate. The last step of the Algorithm, in this work, is to display the results i.e. detection and extraction of all the texts in the RSs which appear in a video. The results may be displayed, for each frame in the video clip, according to one of the following forms:

1- Locating the text on the image by drawing a rectangle around each text, as shown in **Figure (3.17).**

Figure 3.17 Locating the text by bounding box.

2- Crop the texts from the image and display them as shown in **Figure (3.18).**

Figure 3.18 Extracted text of the two road-signs.

Algorithm (3.6) illustrates the steps of detection and extraction of the texts and display the results.

Algorithm (3.6) Detection the Text and Display the results
Input: Cropped RS Image (IM_2), platestartX, platestartY, plateArea **Output:** Display the results

Step 1: Start;
Step 2: Convert(IM_2) to gray-scale image(Imsb);
Step 3: Find t3=CED of (Imsb)with T1=0.04,T2=0.1,δ=3;
Step 4: Apply morphological opening on(t3) with threshold P_2 to find image (g3sb);
Step 5: Apply dilation function with line SE and angle 0°;
Step 6: Apply dilation function with line SE and angle 90°;
Step 7: Apply dilation function with disc SE; //after dilation functions image (Idsb) is obtained//
Step 8: Label all the objects in image (Idsb).

Step 9: Measure the properties of all objects in image(Idsb);

//the properties are: text Height, text Width, the position of the upper-left corner of the object(textstartX, textstartY) and text Area//

Step 10: Measure the number of labels(nsb).

For isb=1:nsb

IF text Width>=1.2 $\times$ $text\ Height$ **and** text Area >0.015 $\times$ $plateArea$ **Then**

Step 11.1 crop object(isb) from image(IM_2);

//coordinates of upper-left corner of object(isb)

= (textstartX,textstartY)

width of object(isb)= textWidth

height of object(isb) = textHeight//

Step 11.2 Draw a rectangle with the following parameter:

- Coordinate of upper-left corner =(platestartX+textstartX,platestartY+textstartY)
- Rectangle width= textWidth
- Rectangle height= textHeight

End IF;

End For;

Step 12: End.

3.9 Arrangement of the Main Algorithm

Majority of the proposed program is described in a form of sub-algorithms [**Algorithms 3.1, 3.2, 3.3, 3.4, 3.5, and 3.6**]. The main algorithm of the program includes all these sub-algorithms taking the following points into account:

1- The following steps are repeated for each frame of the source video:

- **a.** Pre-process the frame. [Algorithm(3.2)]
- **b.** Algorithms (3.3), (3.4) and (3.5) solve the problem of incomplete road-sign plates.
- **c.** Detection of road-sign plates [as described in **Section (3.5.1)**].

The resultant is a detection of the road-signs in the frame.

2- For each road-sign, Algorithm (3.6) is repeated and the final results are obtained.

3- Execution time of the program depends on the following factors:

- **a)** Duration of the video clip (in second).
- **b)** Number of frames per one second.
- **c)** Execution time for one frame, and this time depends on the nature of the frame.

If all frames are processed, the execution time will be long. In video processing applications, it is not required to process all the consecutive frames, because the variation of the scene from frame to the next frame is trivial. In order to reduce the execution time, processing can be carried out on a single frame are a number of frames equals to (frame step). By choosing an appropriate value of (frame step), the objective of the program can be achieved with an acceptable execution time. **Algorithm (3.7)** illustrates the structure of the main algorithm.

Algorithm(3.7) Main Algorithm
Input: Source Video **Output** Detected Text Regions in the Road Signs

Step 1: Start.
Step 2: Call Reading Video File //Algorithm (3.1)//
Step 3:fstep=1; //frame step//
For x=1:fstep:number of frames
Step3.1: Call pre-process the frame; //Algorithm(3.2)//
Step3.2: Display image(IM); //IM=frame-x, this image is appeared as an output with rectangles around the text which is added by Step 11.2 on Algorithm(3.6)//
Step3.3: Apply dilation function with line SE at angle 90̊.
Step3.4: Apply dilation function with line SE at angle 0̊.
//After applying dilation function, image (bw) is obtained//.
Step3.5: Call complete RS which severed from the top of image; //Algorithm(3.3)//

Step4.6: Call complete RS which severed from the left side of image;
//Algorithm(3.4)//
Step3.7: Call complete RS which severed from the right side of image;
//Algorithm(3.5)//
Step3.8: Fill holes in the image;
Step3.9: Label all the objects in the image;
Step3.10: Measure the properties of all objects;
//plateArea, plateWidth, plateHeight, platestartX,platestartY, plateExtent//
Step3.11: Count the number of objects(nobject);
For ii=1:nobject
IF (plateArea>0.01 image area and plateExtent>0.8 and plateWidth/plateHeight is approximately equal to one of the following values[2.25, 3.6, 1.25, 1, and 1.36]) //set of criteria for complete road-sign//
or (plateArea >0.003×image area and plateExtent>0.8 and (object upper corner locate on left **or** top edge of image or upper corner + plateWidth locate on right edge of image**))**//set of criteria for incomplete roadsign//
Then
Step3.12: Crop the object(ii) from image(IM) and save it as an image(IM_2);
// IM is one frame of the source video and IM2 is the image of one of road-signs in IM,

coordinates of upper-left corner of the object(ii) =(platestartX,platestartY),
width of object(ii) =plateWidth,
height of object(ii)=plateHeight //

Step3.13:Call Detection of The text and Display the Results;

// Algorithm(3.6), The output of this algorithm is one of the following forms:

- Display rectangle surrounding the text[Step11.2 in Algorithm(3.6)] on image(IM) [Step4.2].
- Crop and display the texts and save it as an image.//

End IF;

End For;

End For;

Step6: **End.**

Chapter 4

Experimental Results

Chapter 4

Experimental Results

4.1 Evaluation and Measurement the Performance of the Proposed Algorithm

To achieve a comprehensive and accurate evaluation of the work, the algorithm is subjected to several tests, taking into account the different variables and effects.

4.1.1 Evaluation Based on Image Tests

As a first step, to assess the application of the algorithm on video clips, it is necessary to test single frames of these videos. Below, 5 case studies will be reviewed and assessed in detail and later a summary table will show the output results of a number of other cases.

Case Study 1:

- **Description of case study:**

  1. No. of plate in the image: 1.
  2. Color of road-sign background: white.
  3. Font color: black.
  4. Location of the plate: upper.
  5. No. of Arabic words: 5.
  6. No. of English words: 5.
  7. Day time: morning.
  8. Weather condition: clear.
  9. Resolution:1920X1080.

- **The results of case study 1:**

  1. No. of detected Arabic words:5.
  2. No. of detected English words:5.
  3. Total no. of detected words:10.
  4. No. of false positve:0.
  5. No. of missed text:0.
  6. Recall-rate(r):100%.
  7. Precision(p):100%.
  8. Error-rate(ER): 0%.
  9. F-measure(F_s):1.

Figure(4-1) shows the output of applying the algorithm on this case study, the proposed algorithm recognize all the words. Although, shooting of this frame is facing the sun and presence of the plate in shadow region, however detection rate is very high.

-a-

Figure 4.1 **a)** The output of case study 1. **b)**Extracted texts of case study 1.

Case Study 2:

- **Description of case study:**

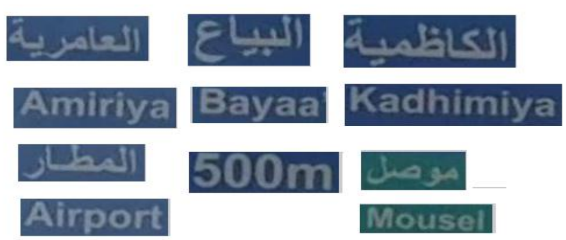
    1. No. of plate in the image:1.
    2. Color of road-sign background: blue.
    3. Font color: white.
    4. Location of the plate: side way.
    5. No. of Arabic words:5.
    6. No. of English words:6.
    7. Day time: noon.
    8. Weather condition: cloudy.
    9. Resolution:1920X1080.

- **The results of case study 2:**

    1. No. of detected Arabic words:5.
    2. No. of detected English words:6.
    3. Total no. of detected words:11.
    4. No. of false positve:0.
    5. No. of missed text:0.
    6. Recall-rate(r):100%.
    7. Precision (p):100%.
    8. Error-rate(ER):0%.
    9. F-measure (Fs):1%.

Figure (4-2) shows the output of applying the algorithm on case study2, the proposed algorithm recognize all the texts. In this frame, all the texts in nearby road-sign plate are detected with no false –positive detection which leads to perfect values of precision and recall rates also, it is noted that there are six far road-sign plates, shooting distance of those plates is more than 120 Meter, so that, their areas appear very small (less than 1% of the image area). Those six road-sign plates are outside the detection zone of the algorithm.

-a-



-b-

Figure 4.2 a) The output of case study 2.**b)** Extracted texts of case study 2.

Case Study 3:

- **Description of case study:**

1. No. of plate in the image: 2.
2. Color of road-sign background: white.
3. Font color: black.
4. Location of the plate: upper.
5. No. of Arabic words:8.
6. No. of English words:8.
7. Day time: noon.
8. Weather condition: clear.
9. Resolution : 1280X720.

- **The results of case study 3:**

1. No of detected Arabic words: 7.
2. No of detected English words:8.
3. Total no. of detected words:15.
4. No. of false positive:0.
5. No. of missed text:1.
6. Recall-rate(r):93.75%.
7. Precision(p):100%.
8. Error-rate(ER):6.25%.
9. F-measure(Fs):0.9674.

Figure(4-3) shows the output of applying the algorithm on case study3, the proposed algorithm does not recognize all the texts, the Arabic word"حي"is not detected because the ratio of its width to its height is not greater than 1.2 which is one of the criteria to consider the object as a text. But, still, the value of r and p are excellent.

-a-

Figure **4.3** **a)** The output of case study 3.**b)** Extracted texts of case study 3.

Case Study 4:

- **Description of case study4:**

  1. No. of plates in the image:2.
  2. Color of road-sign background: blue.
  3. Font color: white.
  4. Location of the plate: upper.
  5. No. of Arabic words:8.
  6. No. of English words:7.
  7. Day time: night.
  8. Weather condition: clear.
  9. Resolution:1920X1080.

- **The results of case study 4:**

  1. No. of detected Arabic words:7.
  2. No. of detected English words:6.
  3. Total no. of detected words:13.
  4. No. of false positive:1.
  5. No. of missed text:2.
  6. Recall-rate(r):86.66%.
  7. Precision (p):92.85%.
  8. Error-rate(er):13.3%.
  9. F-measure (fs):0.894.

Figure (4-4) shows the output of applying the algorithm on the case study4, the proposed algorithm detect most of the texts. This frame is not clear well because the camera is shaking during shooting. However, both of recall and precision values are very good, this gives the impression that the algorithm is successful and reliable even when the vehicle is driven at high speed.

-a-

-b-

Figure 4.4 a) The output of case study 4.**b)** Extracted texts of case study 4.

Case Study 5:

- **Description of case study5:**

  1. No. of plates in the image: 1 (incomplete).
  2. Color of road-sign background: white.
  3. Font color: black.
  4. Location of the plate: upper.
  5. No. of Arabic words:3.
  6. No. of English words:5.
  7. Day time: morning.
  8. Weather condition: clear.
  9. Resolution:1920X1080.

- **The results of case study 5:**

  1. No of detected Arabic words: 3.
  2. No of detected English words:4.
  3. Total no. of detected words:7.
  4. No. of false positive:1.
  5. No. of missed text:1.
  6. Recall-rate(r):87.5%.
  7. Precision (p):87.5%.
  8. Error-rate(ER): 12.5%.
  9. F-measure (Fs):0.875.

Figure (4-5) shows the output of applying the algorithm on case study5. This case study proves that the algorithm works properly with incomplete RSs.

-a-

Figure 4.5 a) The output of case study 5.**b)** Extracted texts of case study 5.

More case studies are analyzed and evaluated and the results are tabulated in **Table (4.1).**

Table (4-1) Results of case study 6-14

Case No.	Resolution	Image Format	Number of Plates	RS. Position	Total No. of Text	Ct	Mt	Ft	r %	p %	Time and weather conditions	Image samples
6	1920x1080	JPEG (.jpg).	1	upper	8	7	1	1	88	88	Morning with shadow	
7	1920x1080	JPEG (.jpg).	2	upper	12	12	0	2	100	86	Noon with bright sun light	
8	1920x1080	JPEG (.jpg).	2	upper	10	7	3	1	70	88	Noon with clouds	
9	1920x1080	JPEG (.jpg).	1	side way	18	14	4	1	78	93	Night with car light	
10	1920x1080	JPEG (.jpg).	1	upper	8	8	0	1	100	89	Noon with sun light reflection	
11	1920x1080	JPEG (.jpg).	1	upper	5	5	0	0	100	100	Noon with clouds	
12	1920x1080	JPEG (.jpg).	1	Single upper	10	9	1	0	90	100	Night with noise	
13	1920x1080	JPEG (.jpg).	1	Single upper	4	4	0	0	100	100	Night with some bluer	
14	1920x1080	JPEG (.jpg).	1	Single upper	6	6	0	0	100	100	Noon with sun light	

The total values of **r** are 89**%** and the total value of **p** are 93**%**. The low results that appeared in the table for the r and p values go to the reason of the distortion that may appear in the video itself, such a distortion a large amount of noise or a shades effects in some cases or the bluer that appears in fast movement of the camera. The incomplete plate example for the single images not included in this table for further example see **Figure (4.5)** . the image sample of **Table(4.1)** are in appendix A.

Figure (4.6) shows the recall and precision performance chart for the image samples in **Section (4.2.1).**

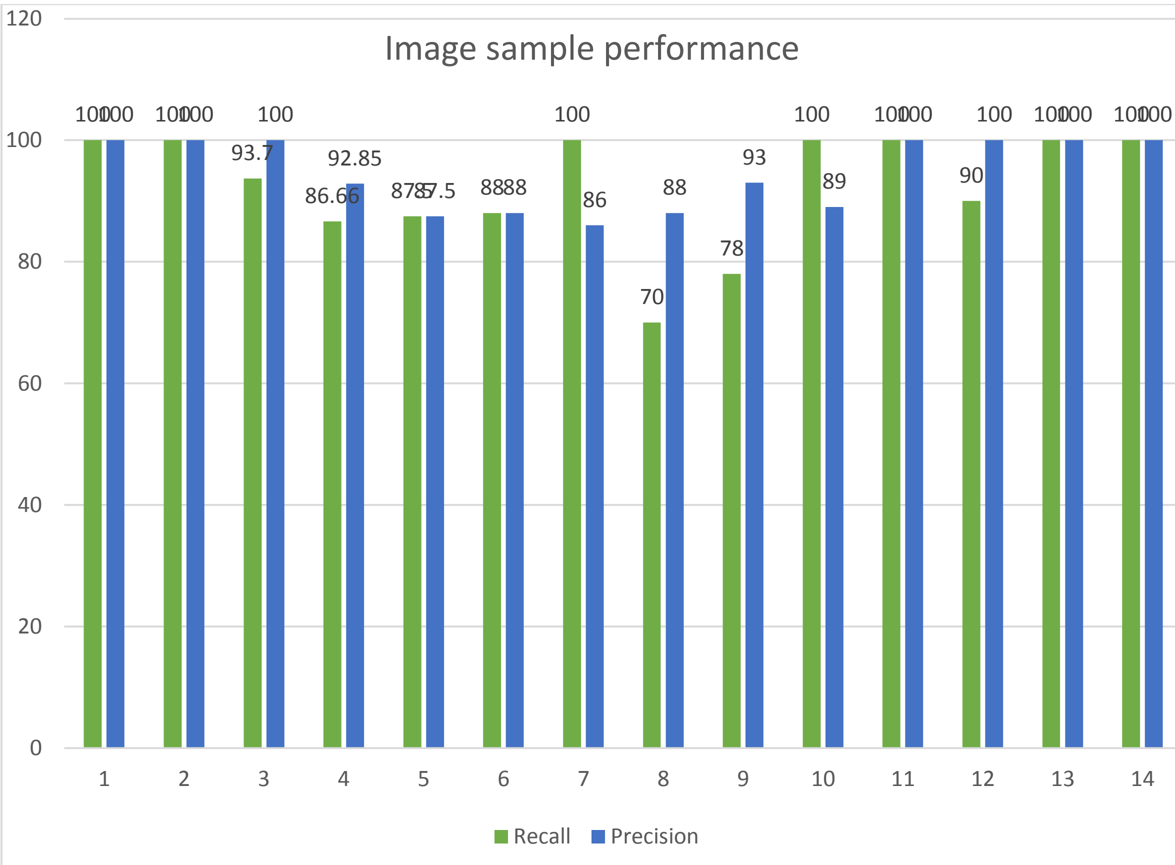


Figure 4.6 Recall and precision for all case studies of image tests.

4.1.2 Evaluation Based on Sequential Fames Tests

Realistic and effective evaluation of the algorithm is achieved by measure its ability to detect texts in video clips. As it is known, the video clip consists of a number of sequential frames. To assess the performance of the algorithm, all the sequential frames of a video are tested and evaluated, the eventual assessment of the algorithm is the resultant of the evaluations of all frames in that video. Below, a video case study will be descripted and measured and later, a summery table will show the evaluation of other videos.

Case Study 1

- **Description of case study1:**

1- Resolution: of the video is 1920x1080 pixel.

2- Format: AVI (.avi).

3- Video duration: 3 Sec.

4- Total number of frames: 72 frame.

5- Day time: noon.

6- Frame rate: 24 frame per second.

7- Weather condition: Dusty weather.

8- Number of words: 6 Arabic and 6 English.

9- Speed of the car: 60km/hr.

In this case study the average recall and average precision is listed in **Table (4-2)**.

Table (4-2) Video sample single case study test results

Case Study	Video Duration In Sec.	Number of Frames	Frames With Text	Frame Without Text	Total No. of Text in all Frame	Ct	Mt	Ft	Recall (%)	Precision (%)	FS	Car Speed	Time of Execution in Sec.
1	3	72	50	22	538	492	46	1	91.4	99.7	0.953	60K/h	162

Figure (4.7) shows recall and precision rates of all frames of the video, but for frames without texts, recall and precision rates cannot be determined.

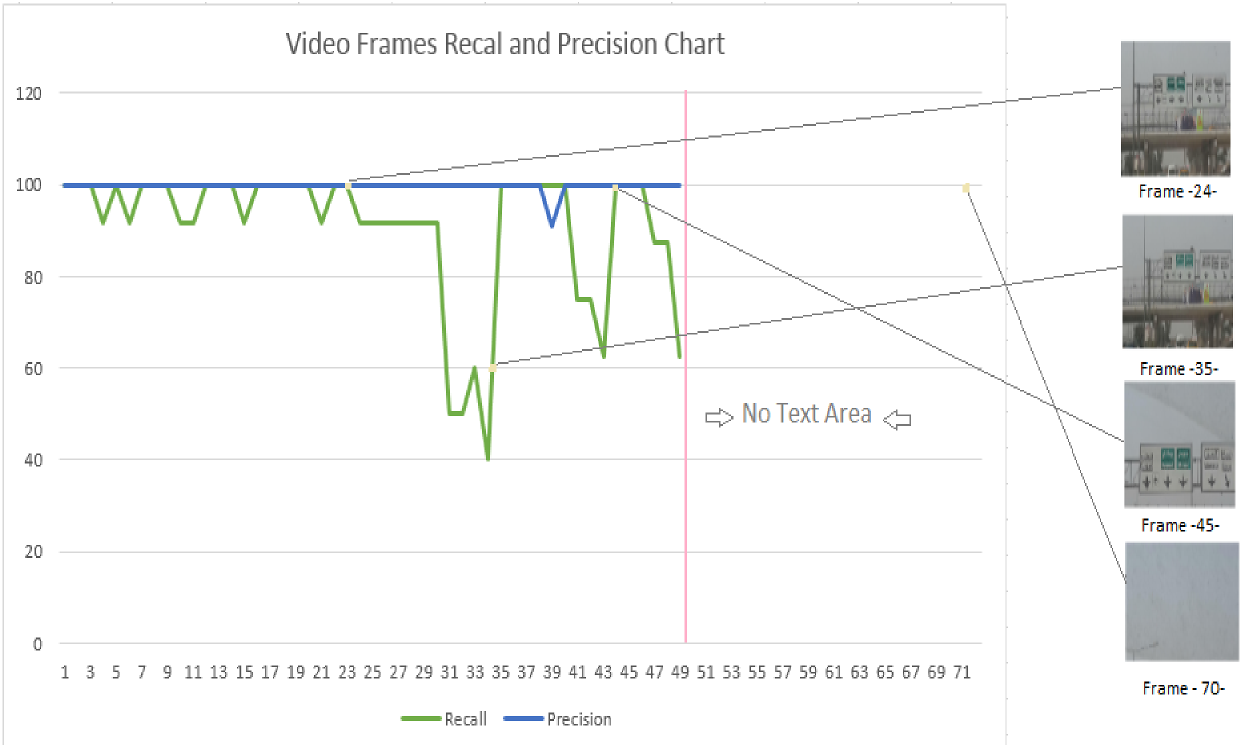


Figure 4.7 Recalls and precisions for sequential frames test.

Obliviously, the values of r and p are changed from frames to another where these values are affected by the following factors:

1- Vibration of the vehicle and the camera.

2- Shooting distance.

3- Shooting angle.

Using the same evaluation approach, another video clips are tested and the results are summarized in **Table (4-3)**.

Table (4-3) Results of sequential frame tests

Case no.	Number of Frames	Frame with Text	Video Duration In Sec.	Frame Rate Per Sec.	Weather Description	Day Time	Total Text Num.	Ct	Mt	Ft	r	p	Execution Time in sec.	Car Speed K/h	FS
2	41	41	1.7	24	Noisy	Night	918	512	406	22	56	96	61.5	60	71
3	68	50	2.8	24	Noisy	Night	400	281	119	191	70	60	91.7	0	65
4	66	66	2.2	30	Shadow	Morning	660	563	97	24	85	96	121.4	0	90
5	29	29	1.16	25	Shadow	Morning	232	207	25	85	89	71	67.3	40	79
6	37	30	1.4	25	Cloudy	Noon	390	319	71	53	82	86	65.4	40	84
7	14	14	0.4	29	sunny	noon	121	86	53	12	71	88	22.7	60	79

In **Table (4-3)** the values of r and p are variant because of the video sample deformation such as large amount of noise or the bluer effects so a low values appear such as in sample 2 in the same table the value of r is 56% because most of the frames contain a bluer effects, that appear most in the night videos and shadow effects videos. Total value of r is 79% and total value of p is 85%. **Figure (4.8)** shows the recall and precision chart for the video samples of **Table (4.3).**

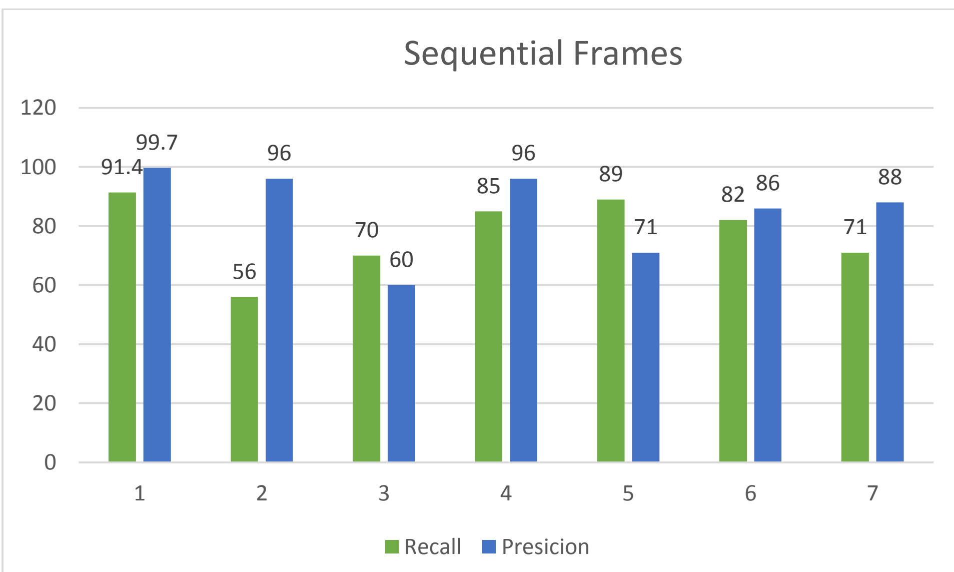


Figure 4.8 Recall and precision for sequential frame tests.

4.1.3 Evaluation Based on Video Tests

In such a sort of applications, it is not necessary to detect the texts of all successive frames. If the algorithm is able to detect all texts which appear during the period of the video clip, this is considered as an excellent indicator in spite of the text detection ratio per each frame. For each tested video, recall and precision rates is determined based on the total number of words, the number of detected words and the number of false-positive detection during that video.

Case Study1

- **Description of case study 1:**

1- Resolution: 1920x1080 pixel.

2- Format: AVI (.avi).

3- Number of Frames: 783 frames.

4- Video duration: 17Sec.

5- Frame Rate:24 frame per second

6- Speed of car: 60 km/hr.

7- Day time: Morning.

8- Weather condition: Sunny.

9- Total Number of Words: 10(5 Arabic words and 5 English words).

- **Results of case study1:**

1- Detected Word: 10.

2- False-positive: 0.

3- Text missed: 0.

4- Recall(r):100%.

5- Precision (p):100%.

6- Error rate(ER):0%.

7- F-measure (Fs):1.

The proposed algorithm extract all texts exist in the video, **Figure (4-9)** shows one frame of the video and the extracted texts from the video.

-a-

–b-

Figure 4.9 a) One frames of the video. **b)** Extracted texts from the video.

Table (4-4) illustrates result of test another case study, it is clear that for all tested cases, the values of recall, precision rates are perfect and this demonstrates and confirms the validity and the accuracy of the proposed algorithm for various operating conditions.

Table (4-4) Results of video text extraction tests

Case No.	Number of Frames	Video Duration In Sec.	Descriptions of the video	Day timey	Total No. of word	ct	mt	ft	r ≈%	p≈%	ER %	Fs
2	134	4	sunny	Noon	8	8	0	5	100	62	0	76
3	306	10	Shadow	Night	12	12	0	2	100	86	0	92
4	236	8	RS slope	Night	28	27	1	5	96.4	84	3.6	90
5	575	19	Vogue vision	Night	18	16	2	6	89	73	11	80
6	333	11	shadow	Morning	10	10	0	2	100	83	0	91
7	43	2	Cloudy	Noon	12	12	0	0	100	100	0	100
8	15	1	Sunny	Morning	4	4	0	4	100	50	0	67
9	47	2	Cloudy	Noon	18	17	1	6	94	74	5.5	83
10	29	2	Cloudy	Noon	11	11	0	0	100	100	0	100

In **Table (4-4)** total value of **r** is 94.5% and **p** is 86.5%, the detection rate of the samples are good most of the 10 samples has reached 100% rate. In sample 8 the precision rate 50% this case is shown a false positive regions because of the video scene has a regions that gives a features such as the text regions .**Figure (4.10)** shows performance chart of **Table (4-4)** samples test.

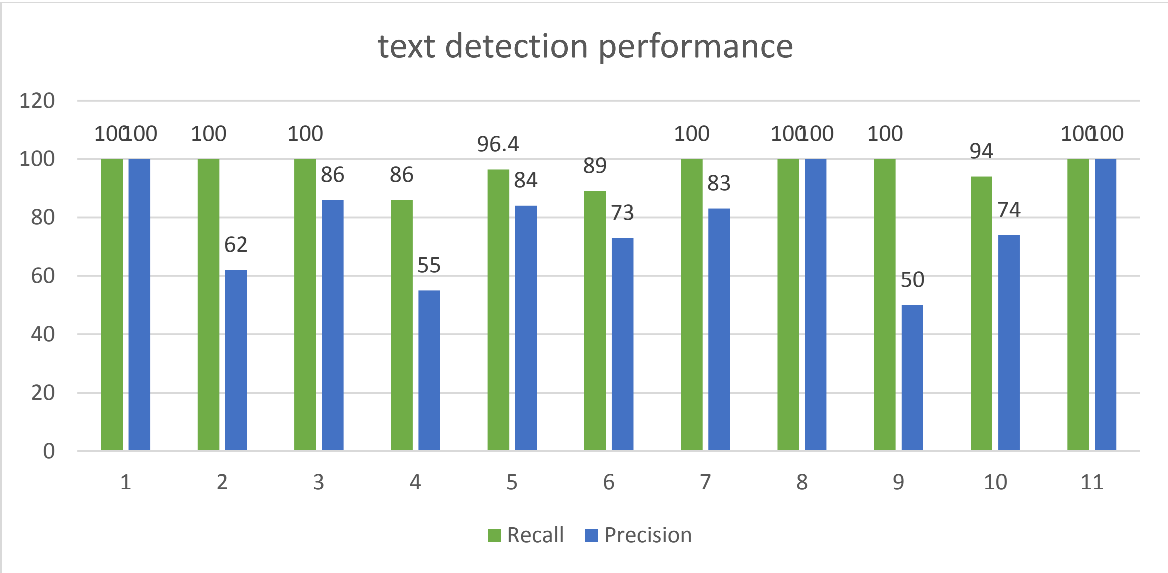


Figure 4.10 Text extraction performance chart.

4.2 Measure the Performance Based on Real-Time Operation

In addition to evaluate the ability of the algorithm to detect the texts in the video clips, there is another important factor which must be measured and discussed, that factor is time required to execute a video clip. It is noted that this time depends on the following parameters:

1- Specification of the video clip.
2- Number of frame per second.
3- Resolution of the video clip.
4- Nature of video clip according to: number of road-signs plates, number of texts, complete or incomplete plate and complexity of background.

By testing the algorithm on a variety of video clips, average execution time of one frame is ranged between **(1.5 Sec. - 2.3 Sec.)**. Since, in videos, the changes that occur on images, from frame to next frame, is insignificant, the algorithm can be formulated to process one frame and exceed a set of next frames. Consequently, the algorithm may operate under real-time condition. To evaluate the suitability of the algorithm to work under real-time conditions, the following case study are tested:

Case study 1

- **Description of case study 1:**

1- Video Duration: 17 Sec.
2- Video Format: Avi (.avi).
3- Total Number of Frames: 422.
4- No. of plates appeared in the video: 4.
5- Total No. of texts: 28.
6- Frame Rate: 24 frame per second.

7- Day time: noon.

8- Weather condition: sunny.

9- Speed of Vehicle: 60 Km/hr.

10- Resolution: this video is converted into three different resolutions (1920×1080),(1280×720),(768×575).

The results of execute this video with the three different resolutions are tabulated in **Table (4-5).**

Table (4-5) Results of real-time operation in case study 1

	Process all Frames of the Video			Process a Reduced Number of Frames		
Resolution	**Total no. of Frames**	**Required Time in Sec.**	**Detection Rate%**	**No. of Processed Frames (percentage)**	**Required Time in Sec.**	**Detection Rate%**
1920×1080	422	360.0	100	9(1:48)	15	93
1280×720	422	209.7	100	18(1:24)	14	89
768×575	422	114.7	96.2	33(1:13)	15	93

Case study 2

- **Description of case study 2:**

1- Video Duration:11.1 Sec.

2- Video Format: Avi(.avi).

3- Total Number of Frames: 267.

4- No. of plates appeared in the video:2.

5- Total No. of text:10.

6- Frame Rate: 24 bit per second.

7- Day time: morning.

8- Weather condition: sunny.

9- Speed of Vehicle: 60 Km/hr.

10- Resolution: this video is converted into three different resolutions (1920×1080),(1280×720),(768×575).

The results of execution of this video with the three different resolutions are tabulated in **Table (4.6).**

Table (4.6) Results of real-time operation in case study 2

	Process all Frames of the Video			Process a reduced Number of Frames		
Resolution	**Total no. of Frames**	**Required Time in Sec.**	**Detection Rate%**	**No. of Processed Frames (percentage)**	**Required Time in Sec.**	**Detection Rate%**
1920×1080	253	416.9	100	6(1:48)	10.2	90
1280×720	254	287.4	100	11(1:24)	9.8	90
768×575	267	88.3	100	13(1:20)	8.1	100

From **Table (4.5)** and **Table (4.6)**, it can be concluded the following notes:

1. When all frames of the video are processed, the detection rate is excellent and this confirms the results obtained in **Section (4.2.3).**
2. To reduce the execution time, the number of processed frames is reduced to insure that the execution time are equal or less than the duration time of the video clip. In this situation, one can say the algorithm is able to work under real-time operation.
3. When the algorithm operates under real-time condition i.e. reduced number of processed frame the detection rates are, still, good and acceptable.

So it could be said that the proposed algorithm has the ability to detect the texts in videos with an acceptable-high detection rate and under real-time condition.

In **Table (4-7)** a Comparison between the proposed work and related work performance.

Tale(4-7) Comparison between the proposed work and related work .

Researchers	**Recall %**	**Precision %**	**Time of Process**	**Night detection**	**Road sign Distance**	**Real time**	**Incomplete plated detection**	**Road-sign Shape**	Road-signs color
Wen W.[6]	88.9	92	Non	No	non	no	no	limited	non
A.VÁZQUEZ[7]	92	67.21	Non	No	short	yes	no	non	Blue or white
Gonz´alez A[8]	92	non	Non	No	short	no	no	non	blue
Jack G.[10]	80	97	Non	No	short	yes	no	limited	non
Proposed Work	94.5	87	1.5-2.2	Yes	long	yes	yes	Different	All Color

Chapter 5

Conclusions and Suggestion for Future Work

Chapter 5

Conclusions and Suggestion for Future Work

5.1 Conclusion

The significant conclusions of this work can be outlined into the following points:

1. Traffic road-signs contain an important and useful information for the road user. Operation of many modern applications, like automated smart vehicle, required an automatic digital recognition the road-sign texts. Discrimination of this text, in images and videos, is a complex task, the reasons of that are: there are many types and forms of the road-signs, movement of the camera and the companied vibrations and the effect of the weather conditions and the lighting level.
2. The road-sign plate may appear, incompletely in some frames of the clip it must be handled so that the texts of those plates can be detected resulting in improving the detection rate of the texts in the video clip.
3. Evaluation based on single image test indicates promising results, where **r** is **89%** and **p** is **93%**.minimum values of text-detection rates appeared at long-distance shooting evaluation based on sequential frames confirmed the success of the algorithm to achieve very good detection-rate. It is not necessary to detect all text of all frames, if the algorithm is able to detect all the words which may appear during the video clip, this is considered as a perfect indicator. Implementation of the algorithm based on video clip give an excellent **recall-rate(r)** with **94.5%** and **precision (P)** with **89.6%.**
4. The algorithm is tested to measure its validity to work under real-time operation, by processing one frame and exceed a set of next frames, the test appears that the algorithm is able to work under real-time operation of **93%.**

5. Car speed has no effect on the detection under a reasonable speed varying between (20-80K/h). A variable name f-step is controlling the frame steps, if the car speed fast(>60k/h) then f-step must be small otherwise f-step must big (example : f-step=10, it means that each 10 frames one frame will be processed).
6. Road-sign plate color has no effect on the detection rate.

5.3 Suggestions for Future Works

The following suggestion can be recommended for future works:

1- Appling text detection, in traffic road-signs, are often a portion of an integrated system which can detect and discriminates text and convert it to understandable and useful information, this work can be evolved in this field.

2- Expanding the scope of this work by using different types of special cameras such as night –vision camera in order to enhance the performance of the algorithm at complete darkness(if needed).

3- Developing an adaptive algorithm to be adapted to the changes in vehicle speed, intensity of illumination and deviation of vehicle on the horizon, and then studying how much improvement occurs to the performance of the algorithm.

4- Developing an algorithm that can analysis the feature of the video scenes to optimize the operation of the camera with regard to shooting direction and zooming in order to improve the text rate.

Reference

[1] H. anoual, D. Aboutajdine, S. Elfkihi Ensias, A. Jilbab Enset, "**Features Extraction for Text Detection and Localization",** IEEE, 2003.

[2] T. Lu ,S. Palaiahnakote, C.L. Tan, and W. Liu **,"Video Text Detection"** ,Springer, 2014.

[3] Jung K, In Kim K, Jain A. K., **"Text information extraction in images and video: a survey",** Elsevier, Volume 37, Issue 5, Pages 977–997, 2004.

[4] Y. Zhong, H.J. Zhang, and A. K. Jain**," Automatic Caption Localization in Compressed Video"**, IEEE, Vol. 22, No. 4, 2000.

[5] Wen W., Xilin Chen, Jie Y., "**Incremental detection of text on road signs**", Carnegie Mellon University,authorized administrator of Research Showcase @ CMU., 2004.

[6] Wen W., Xilin Ch. and Jie Y., "**Detection of Text on Road Signs From Video**", IEEE , VOL. 6, NO. 4, 2005.

[7] A. VÁZQUEZ, R. J. LÓPEZ, S. LAFUENTE, P.GIL **," Adaptive traffic road sign panels text extraction",** WSEAS Int. Conf., pp295-300,2006.

[8] A. Gonz´alez, L.M. Bergasa, J. Javier Yebes and Almaz´an j. ,**" Text Recognition on Traffic Panels from Street-level Imagery"**, IEEE, 2011.

[9] **Jack G. , Majid M.," Recognizing Text-Based Traffic Signs"**, IEEE, Intelligent Transportation Sys., Vol.16, issue 3,2014.

[10] Junhee Youn , Gi H., Kyusoo Chong,**"Automatic extraction of dirction information from road sign images obtained by mobile mapping system"**, Geospatial World Forum,2014.

[11] John W. Woods, "**Multidimensional Signal, Image, and Video Processing and Coding"**, Second Edition, Rensselaer Polytechnic Institute Troy, New York, Elsevier Inc., 2012.

[12] A.L. Bovik**, "Hand Book of Image and video processing",** Academic Press, 2000.

[13] H.R. Wu , K.R. Rao, **"Digital Video Image Quality and Perceptual Coding",** Taylor & Francis Group, 2006.

[14] M. Ebrahim Al-Mualla, C. N. Canagarajah , D. R. Bull **, ”Video Coding for Mobile Communications Effciency, Complexity, and Resilience “,** Elsevier Science (USA),2002.

[15] A. Murat Tekalp**,” Digital Video Processing”,** Prentice Hall Signal Processing Series, Prentice Hall, Englewood CliEs, NJ, 1995.

[16] K. N. Ngan, H. Li, **“Video Segmentation and Its Applications”,** Springer, 2011.

[17] O. Marques, **“Practical Image and Video Processing Using Matlab”**, John Wiley & Sons, 2011.

[18] J. Mukhopadhyay,**” Image and Video Processing in the Compressed Domain”**, Taylor and Francis Group, LLC, 2011.

[19] R. C. Gonzalez, R. E. Woods, **“Digital Image Processing”**, 3rd edition, Prentice Hall,2008.

[20] Scott E. U., **“Digital Image Processing and Analysis: Human and Computer Vision Applications with CVIPtools”,** Taylor and Francis Group, 2010.

[21] Alan C. Bovik,**”The Essential Guide to Image Processing Paperback”**, Elsevier Inc.,2009.

[22] Tinku Acharya, Ajoy K. Ray**, “Image Processing Principles and Applications”,** India, John Wiley & Sons, October, 2005.

[23] Tomas Toss, **“Automatic identification and cropping of rectangular objects in digital images”**, September 2012.

[24] C. Solomon, T. Breckon**, “Fundamentals of Digital Image Processing A Practical Approach with Examples in Matlab**”, John Wiley & Sons, Ltd, 2011.

[25] Frank Y. SHIH**,” Image Processing and Pattern Recognition Fundamentals and Techniques**”, John Wiley & Sons, Inc., 2010.

[26] W. Burger, M. J. Burge,” **Principles of Digital Image Processing Fundamental Techniques”**, Springer, 2009.

[27] Frank Y. Shih**, “Image Processing and Mathematical Morphology: Fundamentals and Applications”,** Taylor and Francis Group, LLC, 2009.

[28] Maria P., **“Image Processing the Fundamentals “,** John Wiley & Sons, University of SurreN Guildford, UK Panagiota Bosdogianni Technical Universify of Crete, Chania, Greece, 1999.

[29] Alasdair M. Andrew, **“An Introduction to Digital Image Processing with Matlab”**, School of Computer Science and Mathematics Victoria University of Technology, Notes for SCM2511 Image Processing, 2004.

[30] J. R. Parker,”**Algorithms for Image Processing and Computer Vision, 2nd Edition”**, John Wiley & Sons, December, 2010.

[31] S. Attaway**, “MATLAB A Practical Introduction to Programming and Problem Solving “**,Second Edition, Elsevier Inc., 2012.

Appendix A

Table (1) Resolution value threshold

Resolution	Sigma1	T1	T2	P1	P2	P3
1920X1080	3	0.04	0.10	130	30	500
1280X720	3	0.04	0.10	130	30	500
768X576	3	0.04	0.10	130	10	50

The value of the thresholds in **Table (1)** are chosen after a long and deep study .the sigma value was determined by studying the effect of sigma on the detected text. The T1 and T2 are determined after sigma value. The value of Pi was determined by studding the area of the possible objects that may appear in the scene of the video, taking by consideration the smallest word may appear in the road sign.

Different video frame studded and under a different conditions the output results was varying detection rate. In **Table (2)** a selected frames with different output is shown to review the results that can be gain from applying the algorithm on a video scenes.

Table (2) Different image test results

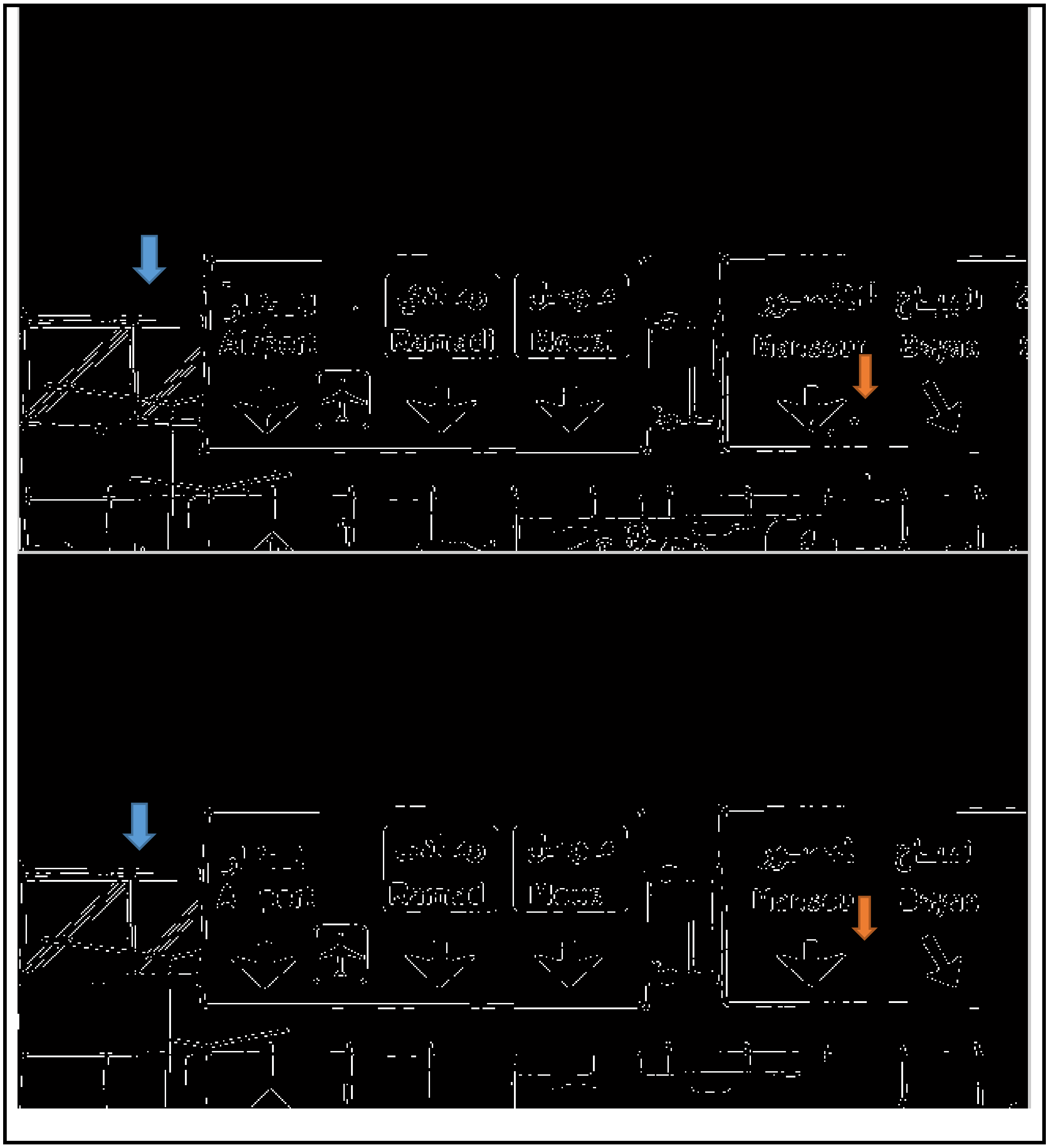
Sample No.	Image Result
1	

2	
3	 Airport Jihad Qu Mousl Kdhmiya 500m Airport Jihad Qu. Amiriya
4	 City Center BEKO Nobel
5	 Doura Bayaa

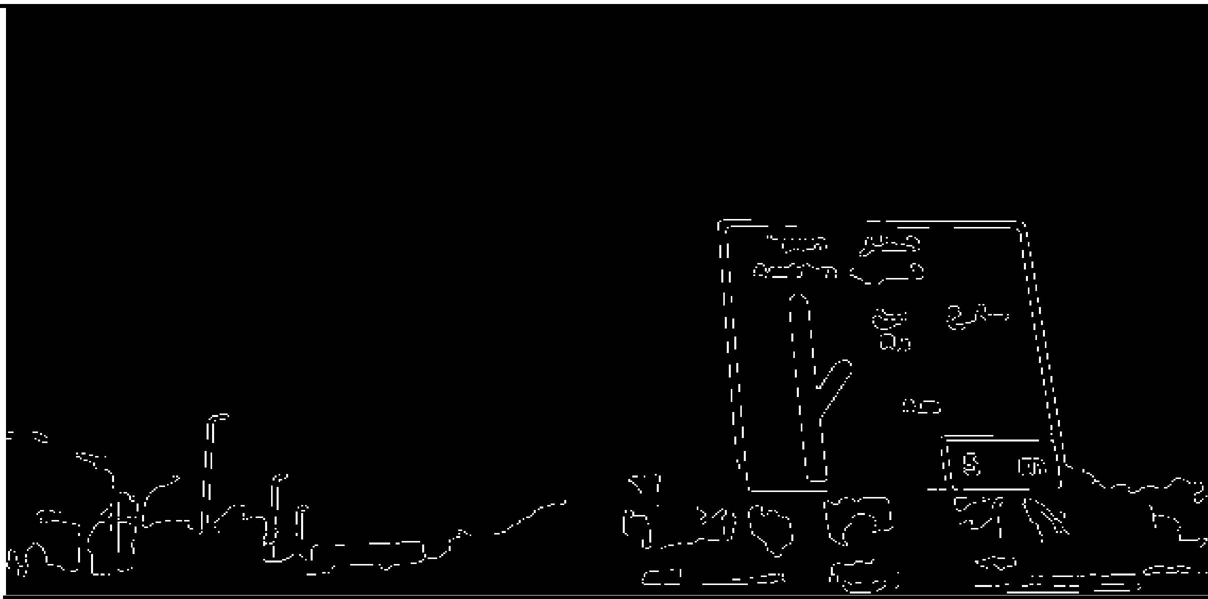
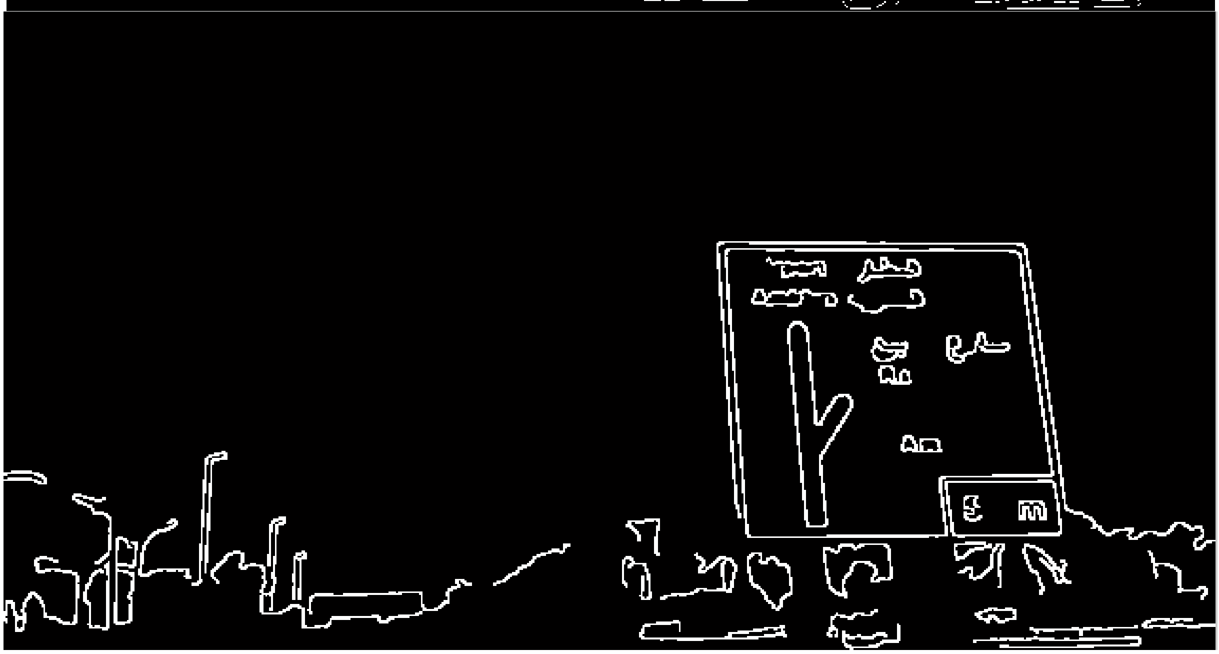
6		 مركز المدينة City Center تقاطع عمان Oman Int. القادسية Qadisiya الدورة Doura البياع Bayaa'
7		 المطار Airport حي الجهاد Jihad Qu. Mousl موصل Ramadi رمادي الكاظمية Kdhmiya
8		 مركز المدينة City Center تقاطع عمان Oman Int. القادسية Qadisiya الدورة Doura البياع Bayaa
9		 المطار Airport رمادي Ramadi موصل Mousl المنصور Mansour البياع Bayaa السيدية Saydiya

10	Airport Jihad Qu City Center Al Qadiseya Highway
11	Airport Amiriya Rabeea St Amel Qu 500m
12	Amel Quarter Airport Nesoor Sq Yarmouk Abu Graib Amriya
13	Technology Univ Zayona Baladiyat NISSAN

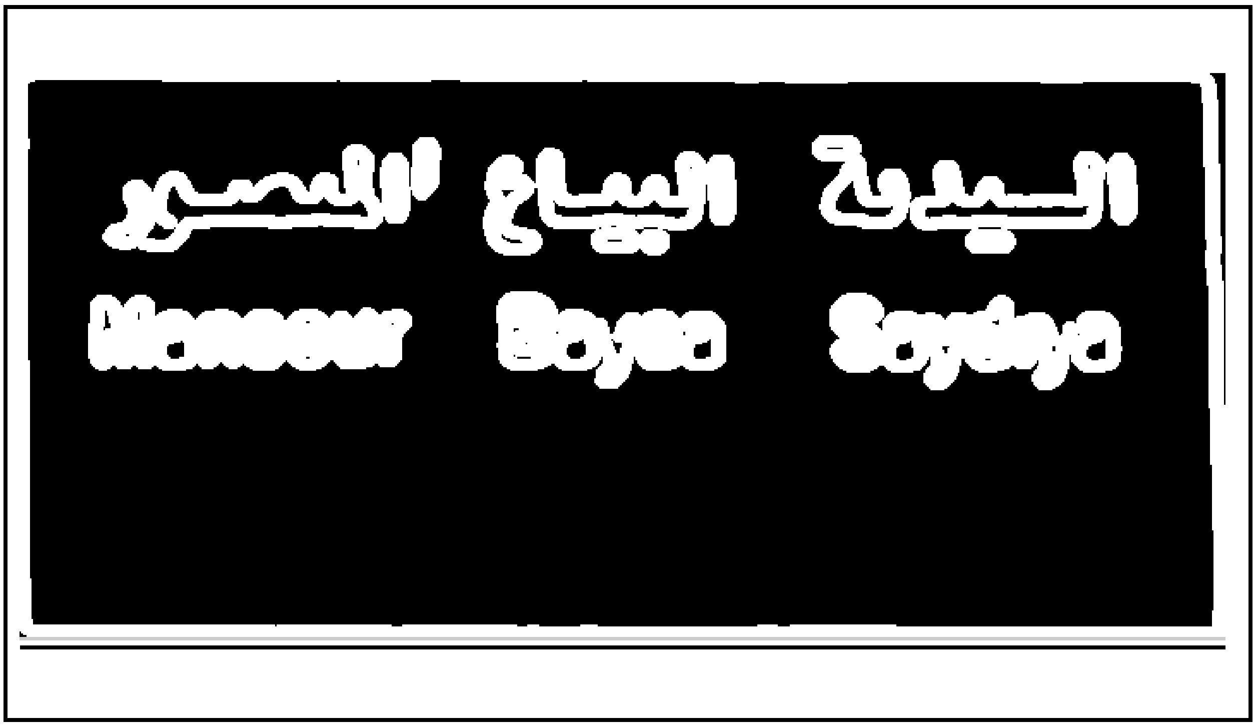
The noise removal step in the algorithm representing by applying the open function the **Example (1)** shows the output of these stages in detail with arrows. **<u>Example (1):</u>**



<u>Example (2):</u> Shows the effects of the applying of the horizontal and vertical Structure Element on the object. In image 1 the output of the canny and the open step ,the image 2 is applying the dilation with horizontal and vertical structure element.

	**Image 1**
	**Image 2**

Example (3): This example shows the effect of applying the horizontal and vertical and disc structure element on the object of an image to link the characters and the dots to gathers and the curvature if exist to get one connected component object. This sample output is the step after applying the final cleaning with p3 threshold.



Example (4): in this example an incomplete road-signs samples are view to show the algorithm performance to solve this problem.


مركز المدينة
City Center
تقاطع عمان
Oman Int.
القادسية
Qadisiya
Karbala - Hilla
الدورة
Doura
البياع
Bayaa'


Bayaa
City Center
Techn Un
Nobel
BEKO


المطار
Airport
رمادي
Ramadi
موصل
Mousl
المنصور
Mansour
البياع
Bayaa
السيدية
Saydiya

جمهورية العراق
وزارة التعليم العالي والبحث العلمي
الجامعة التكنولوجية
قسم علوم الحاسبات

تحليل وتقطيع الفديو الرقمي لتحديد النصوص المكتوبة

رسالة
مقدمة إلى قسم علوم الحاسبات للجامعة التكنولوجية وهي جزء من
متطلبات نيل شهادة الماجستير في برامجيات الحاسوب

من قبل
طيبة ولاء الدين خيري سعيد

بإشراف
أ.د.عبد المنعم صالح رحمة

حزيران- 2015

الخلاصة

تحتوي لوحات الطريق المرورية على معلومات مفيدة لمستخدمي الطريق.عدد من التطبيقات الحديثة مثل المركبات الاوتوماتيكية (الذكية) يتطلب عملها تمييز اوتوماتيكي لنصوص اللوحات المرورية عملية تمييز النصوص تتالف من عدة مراحل ,اول هذه المراحل هو كشف النصوص واستخراجها.في هذا العمل تم تطوير خوارزمية تقوم باكتشاف وتحديد و استخراج النصوص والكتابات من المقاطع الفديوية .تحديدا تهتم هذه الدراسة بالنصوص والكلمات الموجودة في لوحات الطريق المرورية الخاصة بمدينة بغداد حيث تمت دراسة وتحديد خصائص ومميزات هذه اللوحات المرورية.

تمت دراسة قطع الشارع الخاصة بمدينة بغداد وتحديد ابعادها ووضع الموصفات الخاصة بها لتمييزها عن بقية الاجسام الموجودة بالصورة.تم استعمال لغة الماتلاب 2014 لبناء الخوارزمية.

عمل الطريقة المقترحة يتضمن مرحلتين اساسيتين الاولى هي :معالجة الصورة لتحديد موقع وعزل صور اللوحات المرورية واهمال الجزء المتبقي من الصورة,اما المرحلة الثانية هي :معالجة صورة اللوحة المرورية لعزل الكلمات والنصوص دونا عن بقية الرموز والاشكال الهدف من هذا الترتيب هو تقليل زمن التنفيذ اخوارزمية ,البنية الاساسية للخوارزمية تعتمد على الدوال التالية : دالة كاني لكشف الحواف ,دالة التوسيع,دالة ملئ الفجوة,وداله الشكل غير المطوق.

لتقييم اداء وقياس متانة الخوارزمية المقترحة فقد تم اخضاعها لعدة اختبارات حيث تم اختبارعملها على صور مفردة من الفديو وكانت نسبة الاسترجاع(r) في اغلب الاختبارات تساوي او قريبة من 100%,وكانت القيمة الكلية 89% وكذلك نسبة الدقة كانت 93% .

تم اختبار عمل الخوارزمية على مقاطع فديوية اكد امكانياتها الكبيرة في اكتشاف الكلمات التي تظهر في ذللك المشهد حيث كانت نسب الاسترجاع عالية جيدة وكانت القيمة الكلية 94.5%و قيم الدقة 86.5%. تم اختبار صلاحية عمل الخوارزمية تحت الظروف الزمن الحقيقي وذلك بمعالجة صورة واحدة لكل مجموعة من الصور المتعاقبة بينت هذه الاختبارات امكانية الخوارزمية للكشف عن النصوص عند تشغيل الفديو بالزمن الحقيقي وبنسب تحديد 93%.