

**A FULLY AUTOMATED APPLICATION FOR  
ANALYSIS AND QUANTIFICATION OF DNA  
DAMAGE ON COMET ASSAY IMAGES**



**EFTÂL ŞEHİRLİ**

**A FULLY AUTOMATED APPLICATION FOR ANALYSIS AND  
QUANTIFICATION OF DNA DAMAGE ON  
COMET ASSAY IMAGES**

**A THESIS SUBMITTED TO  
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES OF  
KARABUK UNIVERSITY**

**BY**

**EFTÂL ŞEHİRLİ**

**IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR  
THE DEGREE OF PH. D. OF SCIENCE IN  
DEPARTMENT OF  
COMPUTER ENGINEERING**

**September 2018**

I certify that in my opinion the thesis submitted by Eftâl ŞEHİRLİ titled “A FULLY AUTOMATED APPLICATION FOR ANALYSIS AND QUANTIFICATION OF DNA DAMAGE ON COMET ASSAY IMAGES” is fully adequate in scope and in quality as a thesis for the degree of Ph.D. of Science.

Asst. Prof. Dr. Muhammed Kamil TURAN  
Thesis Advisor, Department of Medical Biology



This thesis is accepted by the examining committee with a unanimous vote in the Department of Computer Engineering as a Ph.D. thesis. September 17, 2018

Examining Committee Members (Institutions)

Signature

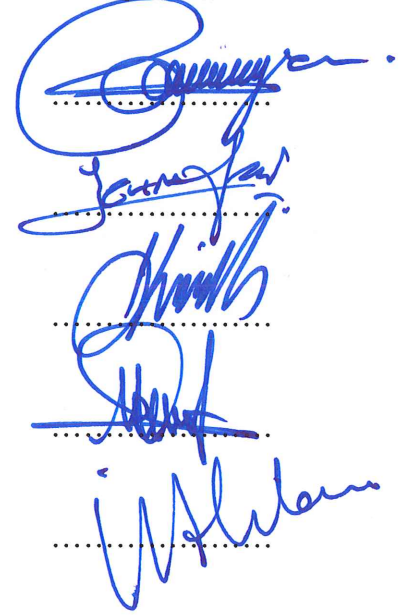
Chairman : Prof. Dr. Bülent BAYRAM (YTU)

Member : Assoc. Prof. Dr. Zehra SAFİ ÖZ (BEU)

Member : Asst. Prof. Dr. Muhammed Kamil TURAN (KBU)

Member : Asst. Prof. Dr. Mehmet KARA (KBU)

Member : Asst. Prof. Dr. Ümit ATİLA (KBU)



..... / ..... / 2018

The degree of Ph.D. of Science by the thesis submitted is approved by the Administrative Board of the Graduate School of Natural and Applied Sciences, Karabük University.

Prof. Dr. Filiz ERSÖZ  
Head of Graduate School of Natural and Applied Sciences





*“I declare that all the information within this thesis has been gathered and presented in accordance with academic regulations and ethical principles and I have according to the requirements of these regulations and principles cited all those which do not originate in this work as well.”*

Eftâl ŐEHİRLİ

## **ABSTRACT**

**Ph. D. Thesis**

# **A FULLY AUTOMATED APPLICATION FOR ANALYSIS AND QUANTIFICATION OF DNA DAMAGE ON COMET ASSAY IMAGES**

**Eftâl ŞEHİRLİ**

**Karabük University**

**Graduate School of Natural and Applied Sciences**

**Department of Computer Engineering**

**Thesis Advisor:**

**Asst. Prof. Dr. Muhammed Kamil TURAN**

**September 2018, 100 Pages**

Increasingly, for many biomedical areas, it is becoming more important to develop a software application that automatically quantifies and presents parametric results for users. Single cell gel electrophoresis known as comet assay is one of the most preferred methods in biomedical areas which users need to obtain parametric results about DNA damage. An efficient and detailed analysis and quantification on comet assay images has been performed by a fully automated application developed using Python programming language in this thesis study. It is separately and respectively performed on comet assay images to extract comet objects, eliminate non-comet objects like small, blurry and overlapped, grade damage level such as healthy, mild, medium or severe, and calculate comet parameters such as comet length, comet area, head length, head area, head percentage, tail length, tail area, tail percentage and tail moment. A novel thresholding method has been developed to convert grayscale images to binary

images. A novel method based on signal processing and pattern recognition has been developed to eliminate overlapped comet objects. Besides, a novel method based on pixel profile analysis, dynamic time warping and decision tree has been developed to grade damage level. 2476 comet assay images captured at the end of comet assay experimental studies have been used in this thesis study. 72.22% for sensitivity, 93.33% for specificity and 81.82% for accuracy are obtained to eliminate blurry objects. 91.30% for sensitivity, 93.24% for specificity and 92.88% for accuracy are obtained to eliminate overlapped comet objects. Sensitivity ranging between 72.22% and 97.22%, specificity ranging between 88.89% and 100% and accuracy ranging between 90.21% and 99.30% are obtained to grade damage level. The developed application has been compared with OpenComet and Comet IV applications using same comet assay images.

**Key Words** : Comet assay, DNA damage, head part of comet, tail part of comet, image processing.

**Science Code** : 924.1.014

## ÖZET

**Doktora Tezi**

### **KUYRUKLU YILDIZ GÖRÜNTÜLERİNDE DNA HASARI ÖLÇÜMÜ VE ANALİZİ İÇİN TAM OTOMATİK UYGULAMA**

**Eftâl ŞEHİRLİ**

**Karabük Üniversitesi**

**Fen Bilimleri Enstitüsü**

**Bilgisayar Mühendisliği Anabilim Dalı**

**Tez Danışmanı:**

**Dr. Öğr. Üyesi Muhammed Kamil TURAN**

**Eylül 2018, 100 Sayfa**

Birçok biyomedikal alan için, kullanıcılara otomatik olarak ölçüm yapan ve parametrik sonuçlar veren yazılım uygulamalarını geliştirmek giderek daha önemli hale gelmektedir. Kuyruklu yıldız analizi olarak bilinen tek hücre jel elektroforezi, kullanıcıların DNA hasarı hakkında parametrik sonuçlar elde etmek için ihtiyaç duydukları biyomedikal alanlarda en çok tercih edilen metotlardan biridir. Bu tez çalışmasında Python programlama dili kullanılarak geliştirilen tam otomatik çalışan uygulama ile kuyruklu yıldız görüntüleri üzerinde verimli ve detaylı analiz ve ölçüm yapılmıştır. Ayrı ayrı ve sırasıyla kuyruklu yıldız görüntüleri üzerinde kuyruklu yıldız nesnelere çıkarılma, küçük, bulanık, çakışık gibi kuyruklu yıldız olmayan nesnelere eleme, hasar seviyelerini sağlıklı, hafif, orta ve ağır olarak derecelere ayırma ve kuyruklu yıldız uzunluk, kuyruklu yıldız alan, baş kısmı uzunluk, baş kısmı alan, baş kısmı yüzde, kuyruk kısmı uzunluk, kuyruk kısmı alan, kuyruk kısmı yüzde ve kuyruk

momenti olarak parametrelerin hesaplama işlemi yapılmıştır. Gri seviye resimleri ikili resimlere dönüştürmek için özgün bir eşikleme metodu geliştirilmiştir. Çakışık kuyruklu yıldız nesnelere elemek için sinyal işleme ve örüntü tanıma tabanlı özgün bir metot geliştirilmiştir. Bunun yanında, hasar seviyelerini sınıflara ayırmak için de piksel profil analizi, dinamik zaman bükme ve karar ağacı tabanlı özgün bir metot geliştirilmiştir. Kuyruklu yıldız analizi deney çalışmaları sonunda 2476 adet kuyruklu yıldız analiz görüntüleri çekilmiştir. Bulanık nesnelere elemelerde %72.22 duyarlılık, %93.33 özgüllük ve %81.82 doğruluk elde edilmiştir. Çakışık kuyruklu yıldız nesnelere elemelerde %91.30 duyarlılık, %93.24 özgüllük ve %92.88 doğruluk elde edilmiştir. Hasar dereceleri ayırmada %72.22 ile %97.22 arasında değişen duyarlılık, %88.89 ile %100 arasında değişen özgüllük ve %90.21 ile %99.30 arasında değişen doğruluk elde edilmiştir. Geliştirilen uygulama aynı kuyruklu yıldız görüntüleri kullanılarak OpenComet ve Comet IV uygulamaları ile karşılaştırılmıştır.

**Anahtar Sözcükler :** Kuyruklu yıldız analizi, DNA hasarı, kuyruklu yıldız baş kısmı, kuyruklu yıldız kuyruk kısmı, sayısal görüntü işleme.

**Bilim Kodu :** 924.1.014



## ACKNOWLEDGMENT

Foremost, I owe my deepest gratitude to my thesis advisor Asst. Prof. Dr. Muhammed Kamil TURAN for his help, support, interest, experiences throughout this thesis study.

Throughout this thesis study, I am deeply indebted to Asst. Prof. Dr. Mehmet KARA for his knowledge, supports and efforts especially in comet assay experiments.

I am highly appreciated to my family owing to give their invaluable efforts in brief.

Throughout my Ph.D. years, I wish to express my warmest appreciation to Anday DURU and Emrullah DEMİRAL for motivation support and contribution of nice memories.

I desire to thank our scientific and academic team members Abdullah ELEN, Hakan YILMAZ, Murat KORKMAZ and Berk ŞAHİN for unforgettable information exchange and memories.

I would like to present thanks Assoc. Prof. Dr. Zehra SAFİ ÖZ and Assoc. Prof. Dr. Meryem AKPOLAT owing to perform comet assay experiments together in Comet Assay laboratory of Bülent Ecevit University.

The project was supported by Karabük University with KBÜ-BAP-16/2-DR-102 project number. I want to present my thanks to Karabük University.

## CONTENTS

	<u>Page</u>
APPROVAL.....	ii
ABSTRACT.....	iv
ÖZET.....	vi
ACKNOWLEDGMENT.....	viii
CONTENTS.....	ix
LIST OF FIGURES .....	xii
LIST OF TABLES .....	xv
SYMBOLS AND ABBREVIATIONS INDEX.....	xvi
CHAPTER 1 .....	1
INTRODUCTION .....	1
CHAPTER 2 .....	4
COMET ASSAY.....	4
CHAPTER 3 .....	11
PYTHON AND FUNDAMENTAL PACKAGES .....	11
3.1. PYTHON.....	11
3.2. PYTHON FUNDAMENTAL PACKAGES .....	12
3.2.1. NumPy.....	12
3.2.2. SciPy.....	13
3.2.3. SciKits .....	13
3.2.4. Python Imaging Library (PIL).....	13
3.2.5. Matplotlib .....	13
3.2.6. OpenCV-Python .....	14
CHAPTER 4 .....	15
LITERATURE REVIEW.....	15

	<u>Page</u>
CHAPTER 5 .....	25
UNIFIED MODELLING LANGUAGE DIAGRAM OF THE DEVELOPED APPLICATION .....	25
5.1. USE CASE DIAGRAM .....	26
5.1.1. Use Case 1 .....	27
5.1.2. Use Case 2 .....	28
5.1.3. Use Case 3 .....	28
5.2. RELATIONSHIPS BETWEEN USE CASES .....	28
5.3. ACTIVITY DIAGRAM .....	29
5.4. REQUIREMENT DIAGRAM .....	31
5.5. COMMUNICATION DIAGRAM .....	32
5.6. RELATIONSHIPS BETWEEN OBJECTS .....	33
5.6.1. Association .....	33
5.6.2. Aggregation .....	34
5.6.3. Composition.....	34
5.6.4. Generalization and Specialization .....	35
CHAPTER 6 .....	37
MATERIALS .....	37
CHAPTER 7 .....	40
METHODS .....	40
7.1. PREPROCESSING STAGE .....	41
7.1.1. Gaussian Filter .....	41
7.1.2. Median Filter .....	43
7.1.3. Thresholding Method.....	43
7.2. SEGMENTATION STAGE.....	45
7.2.1. Connected Component Labeling .....	45
7.2.2. Removing Objects from Image Border.....	46
7.2.3. Extracting Individual Objects .....	47
7.3. ELIMINATION STAGE.....	47
7.3.1. Elimination of Small Objects.....	47
7.3.2. Elimination of Blurry Objects .....	47

	<u>Page</u>
7.3.2.1. Variance of Laplacian Method.....	48
7.3.2.2. Entropy of Histogram Method .....	49
7.3.2.3. Gradient Energy Method.....	49
7.3.3. Elimination of Overlapped Comets .....	50
7.4. ANALYSIS STAGE .....	54
7.4.1. Detection of Center of Head Part.....	54
7.4.2. Detection of Tail Direction .....	54
7.4.3. Separating Head Part from Tail Part.....	55
7.4.4. Calculation of Comet Parameters .....	55
7.4.5. Grading Damage Level.....	57
7.4.5.1. Pixel Profile Analysis .....	57
7.4.5.2. Dynamic Time Warping (DTW).....	57
7.4.5.3. Decision Tree .....	62
7.4.5.4. Measurement Parameters of Decision Tree .....	65
7.5. PRESENTING RESULTS STAGE .....	67
7.6. VALIDATION .....	71
CHAPTER 8 .....	73
RESULTS .....	73
8.1. ELIMINATION RESULTS .....	74
8.1.1. Elimination of Small Objects.....	74
8.1.2. Elimination of Blurry Objects .....	74
8.1.3. Elimination of Overlapped Comets .....	79
8.2. GRADING RESULTS .....	82
CHAPTER 9 .....	90
CONCLUSION.....	90
REFERENCES.....	92
APPENDIX A. GUI OF THE DEVELOPED APPLICATION .....	98
RESUME .....	100

## LIST OF FIGURES

	<u>Page</u>
Figure 2.1. An individual comet object and parts of it. ....	5
Figure 2.2. Comet objects for each damage level.....	5
Figure 2.3. Processes of comet assay experiments adapted from [17]. ....	8
Figure 2.4. Area of head part. ....	9
Figure 2.5. Center of head part. ....	9
Figure 2.6. Radius of head part.....	10
Figure 2.7. Area of tail part. ....	10
Figure 2.8. Length of tail part.....	10
Figure 5.1. Use case diagram.....	27
Figure 5.2. Relationships between use cases. ....	29
Figure 5.3. Activity diagram of analyzing comet objects.....	30
Figure 5.4. Activity diagram of capturing an image from camera. ....	31
Figure 5.5. Requirement diagram. ....	32
Figure 5.6. Communication diagram of analyzing an image.....	32
Figure 5.7. Communication diagram of capturing and analyzing an image.....	33
Figure 5.8. Communication diagram of applying image processing techniques.....	33
Figure 5.9. The representation of Association relationship. ....	34
Figure 5.10. The representation of Aggregation relationship.....	34
Figure 5.11. The representation of Composition relationship. ....	35
Figure 5.12. The representation of Generalization and Specification relationship. ..	35
Figure 5.13. Object diagram. ....	36
Figure 6.1. Olympus CX31 trinocular microscope.....	38
Figure 7.1. The flow chart of the algorithm.....	40
Figure 7.2. Comet assay image and histograms of channels ....	41
Figure 7.3. Calculation process of mean value of first row in M in cascade and binomial way. ....	44
Figure 7.4. 8-neighbors connectivity. ....	46
Figure 7.5. Laplace kernel to perform Eq. 7.10 and adapted forms. ....	49
Figure 7.6. The flow chart of elimination of overlapped comets. ....	50
Figure 7.7. Pattern samples.....	53

	<u>Page</u>
Figure 7.8. Individual comet object sample.....	58
Figure 7.9. DTW results for the sequences X and Y.....	59
Figure 7.10. Possibilities of the nearest point for i-th element.....	60
Figure 7.11. Path between the sequences X and Y.....	61
Figure 7.12. Diagonal path and path obtained by DTW.....	66
Figure 7.13. A sample view of a folder.....	68
Figure 7.14. A sample loaded comet assay image.....	69
Figure 7.15. A sample excel file for a loaded comet assay image.....	69
Figure 7.16. A sample individual comet object.....	70
Figure 7.17. A sample pdf file for an individual comet object.....	70
Figure 8.1. Decision tree structure obtained by variance of Laplacian with 2-level pruning.....	75
Figure 8.2. Decision tree structure obtained by entropy of histogram with 2-level pruning.....	75
Figure 8.3. Decision tree structure obtained by gradient energy with 4-level pruning.....	76
Figure 8.4. Decision tree structure obtained by multiplication of variance of Laplacian with entropy of histogram with 3-level pruning.....	76
Figure 8.5. Decision tree structure obtained by multiplication of entropy of histogram with gradient energy with 3-level pruning.....	77
Figure 8.6. Decision tree structure obtained by multiplication of variance of Laplacian with gradient energy with 2-level pruning.....	77
Figure 8.7. Decision tree structure obtained by multiplication of all with 4-level pruning.....	78
Figure 8.8. Decision tree structure with 1-level pruning for elimination of blurry comet objects.....	79
Figure 8.9. A sample overlapped comet, its lane histogram and its smoothed lane histogram.....	80
Figure 8.10. A sample overlapped comet, its lane histogram and its smoothed lane histogram.....	80
Figure 8.11. A sample overlapped comet, its lane histogram and its smoothed lane histogram.....	80
Figure 8.12. A sample overlapped comet, its lane histogram and its smoothed lane histogram.....	80
Figure 8.13. A sample overlapped comet, its lane histogram and its smoothed lane histogram.....	81
Figure 8.14. ROC graph of seven different window sizes of moving average filter.....	81

	<b><u>Page</u></b>
Figure 8.15. G3 comet object.....	83
Figure 8.16. G2 comet object.....	84
Figure 8.17. G1 comet object.....	85
Figure 8.18. G0 comet object.....	86
Figure 8.19. Decision tree structure to grade damage level.....	87
Figure Appendix A.1. GUI of the developed application. ....	99



## LIST OF TABLES

	<u>Page</u>
Table 4.1. Number of publications versus year.....	16
Table 4.2. Comparison of publications according to sensitivity.....	23
Table 4.3. Image type, application type and the used methods.....	23
Table 6.1. Comet assay materials.....	37
Table 6.2. Properties of Olympus E-330 pro camera.....	38
Table 6.3. Properties of the used computer.....	39
Table 7.1. Some pixel intensity values of an individual comet object sample. ....	58
Table 7.2. Cost matrix for the sequences X and Y.....	59
Table 7.3. Confusion matrix.....	72
Table 8.1. Validation of parameters for elimination of blurry objects.....	78
Table 8.2. Validation for seven different window sizes of moving average filter.....	81
Table 8.3. The comparison between OpenComet and the developed application. ....	82
Table 8.4. Measurement parameters of each comet object in Figure 8.15-8.18. ....	86
Table 8.5. Interval of six measurement parameters for each damage level. ....	87
Table 8.6. Confusion matrix based on accuracy for each damage level. ....	88
Table 8.7. Confusion matrix based on sensitivity and specificity for each damage level. ....	89



## SYMBOLS AND ABBREVIATIONS INDEX

### SYMBOLS

- °C : Degree Celsius  
 $\sigma$  : Standard Deviation

### ABBREVIATIONS

- ACHP : Abscissas of Center of Head Part  
CART : Classification and Regression Trees  
CM : Centimeter  
CPU : Central Processing unit  
DMSO : Dimethyl Sulfoxide  
DNA : Deoxyribonucleic Acid  
DPBS : Dulbecco's Phosphate Buffered Saline  
DTW : Dynamic Time Warping  
FN : False Negative  
FP : False Positive  
G0 : Healthy  
G1 : Mild  
G2 : Medium  
G3 : Severe  
GB : Giga Byte  
GHz : Giga Hertz  
GUI : Graphical User Interface  
ID3 : Iterative Dichotomiser 3  
LMPA : Low Melting Point Agarose  
MA : Milliampere  
MATLAB : Matrix Laboratory

ML	: Milliliter
MB	: Mega Byte
MHz	: Mega Hertz
NMPA	: Normal Melting Point Agarose
OCHP	: Ordinate of Center of Head Part
OMT	: Object Modelling Technique
OOAD	: Object Oriented Analysis & Design
OOSE	: Object Oriented Software Engineering
PDF	: Portable Document Format
PIL	: Python Imaging Library
PH	: Power of Hydrogen
RAM	: Read Access Memory
RMSE	: Root Mean Square Error
ROC	: Receiver Operating Characteristic
ROI	: Region of Interest
SCIKITS	: SciPy Toolkits
SRS	: System Requirement Specification
SVM	: Support Vector Machine
TIFF	: Tagged Image File Format
TN	: True Negative
TP	: True Positive
UML	: Unified Modelling Language
V	: Volt
$\mu$ L	: Microliter

## CHAPTER 1

### INTRODUCTION

Medical imaging and software applications that automatically perform analysis in biomedical area have increased their popularities in recent years. Both substantially provide convenience for doctors, scientists and users who are interested in medicine field. Combinatorial systems which include hardware and software together have a key role to present information about diseases, experimental measures, damages etc.

In last years, applications in biomedical area have started to be developed in a fast way. Applications usually show abnormal features, diseases, damages and present numerical parameters about them using images captured by devices like microscope and camera. Quality of images is very important to display these features, perform analysis on images and obtain results in an accurate manner.

Deoxyribonucleic acid (DNA) is one of the necessary topics worked in genetic area. DNA is an important structure such that genetic information is healthily transferred from generation to generation due to DNA. However, DNA that carries genetic information is a molecule which can easily damage. Etiology of many diseases like cancer, cardiovascular disease, and immune deficiency is asserted to be based on DNA damage [1,2]. DNA damage is measured as an indispensable parameter such that various experimental methods have had a part in measuring DNA damage [2,3]. In particular, single cell gel electrophoresis or shortly named as comet assay is a fast, cheap, sensitive and reliable method which can detect DNA damage and identify damage level in eukaryotic cells or tissues. If DNA has damage, DNA is stained in the gel after electrophoresis. Thus, it appears like a comet shape and is observed with the help of microscopes. Therefore, comet assay, one option, is utilized to detect damage on DNA (single strand or double strand breaks) [4-8].

Python is a developing programming language recently. Python presents distinctive and effective toolboxes like Numerical Python (NumPy), Scientific Python (SciPy), biopython especially for scientific developments and measurements. Python is frequently used in biology, bioinformatics and biomedical image processing areas.

Even though detection of DNA damage is very important criteria, there are few studies about automated detection, analysis and quantification of DNA damage in the literature. Four studies about this subject semi-automatically detects DNA damage. They need user interactions before giving DNA damage results. No studies eliminate both blurry and overlapped comet objects to obtain all true comet objects before analyzing DNA damage. That's why, in this thesis study, it is aimed to develop a fully automated application that analyzes and quantifies DNA damage by using novel and unstudied methods with a high success rate.

In this thesis study, it is aimed that all true comet objects that can be analyzed are detected. Comet objects at borders and non-comet objects like small, blurry and overlapped cannot be analyzed in a correct way since their morphologies are not obvious and understandable. Moreover, non-comet objects make analysis accuracy and performance decrease. Thus, when all true comet objects are obtained, and non-comet objects are eliminated, all true comet objects are quantified and analyzed.

This thesis study mainly introduces a fully automated application developed using Python programming language to analyze and quantify comet assay images. It is separately and respectively performed on comet assay images to extract individual comet objects, eliminate non-comet objects like small, blurry and overlapped, grade damage level such as healthy (G0), mild (G1), medium (G2) or severe (G3) and calculate comet parameters such as comet length, comet area, head length, head area, head percentage, tail length, tail area, tail percentage and tail moment. RMSE and damage level are new comet parameters added to the literature. Three novel methods have been developed and used in the application. First one is a thresholding method to convert grayscale images to binary images. Second one based on signal processing and pattern recognition has been developed to eliminate overlapped comets. Third one

based on pixel profile analysis, dynamic time warping (DTW) and decision tree has been developed to grade DNA damage level.

The developed application presents results related to comet parameters and DNA damage level. However, it cannot appoint any medical diagnosis according to presented results. The developed application can analyze only comet assay images stained with Ethidium bromide. Comet assay images should be captured by a camera mentioned in Chapter 6 and using 40X ocular objective lens. There is a possibility as obtaining results may change when another camera and another objective lens size are used.

This thesis study is organized in 9 chapters. In Chapter 1, introduction is described. In Chapter 2, comet assay method is described. In Chapter 3, Python programming language and its packages are described. In Chapter 4, the literature review corresponding to comet assay analysis applications and methods are described. In Chapter 5, Unified modelling language (UML) models of the developed application are described. In Chapter 6, used materials during both comet assay experiments and development of the application are described. In Chapter 7, developed methods are explained. In Chapter 8, obtained results are presented. In Chapter 9, the thesis study is concluded.

## CHAPTER 2

### COMET ASSAY

Single cell gel electrophoresis or shortly comet assay is a standard method to identify DNA damage in individual cells. The idea of single cell electrophoresis to measure DNA damage was introduced by Rydberg and Johanson [4,5]. Then, the idea was named as comet assay in 1984 by developing the method by Östling and Johansson [4,6]. Singh et al. introduced to modified version of comet assay which included alkaline conditions in 1988 [4,7]. Scope of comet assay method is quite large and areas of its usage are fundamental research in DNA damage and repair, human biomonitoring and molecular epidemiology, diagnosis of genetic disorders, monitoring environmental contamination with genotoxins and testing novel chemicals for genotoxicity [8].

Comet assay is used to measure single or double strand DNA breaks, alkaline fragile regions, DNA cross-links and apoptotic nuclei in cells. DNA which has damage is stained in a gel and it appears like a comet. DNA which does not have any damage is stained in a gel, but it does not appear like a comet. It appears like smooth round. Thus, this technique is called as comet assay [4-8]. An individual comet object and its parts are shown in Figure 2.1.

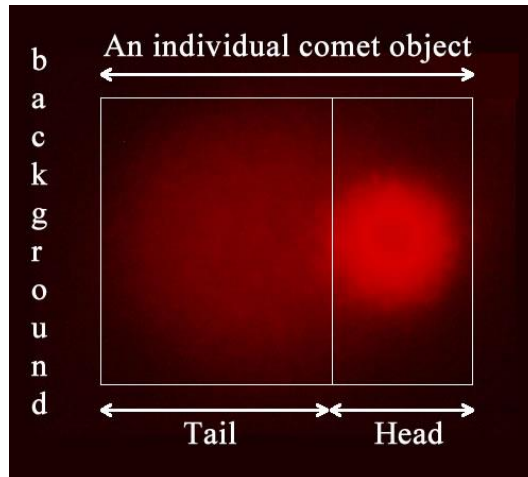


Figure 2.1. An individual comet object and parts of it.

In this thesis study, comet objects are classified according to damage level such as G0 as healthy, G1 as mild damage level, G2 as medium damage level and G3 as severe damage level. Comet objects for each damage level are shown in Figure 2.2.

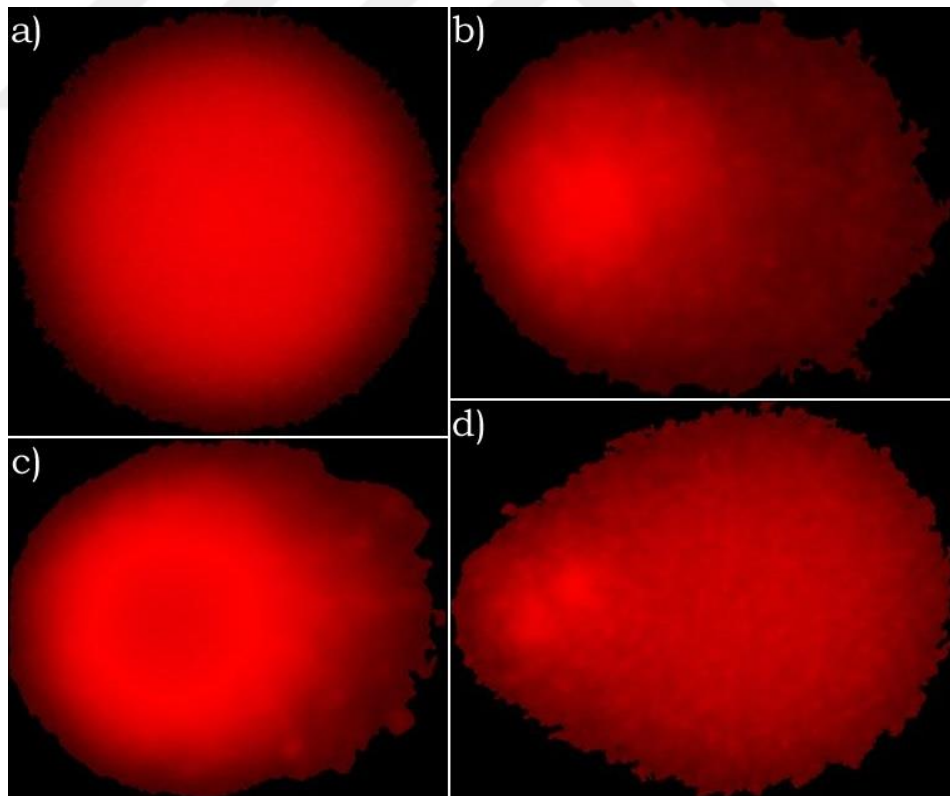


Figure 2.2. Comet objects for each damage level. a) G0 comet object, b) G2 comet object, c) G1 comet object, d) G3 comet object.

In this thesis study, comet assay experiments have been performed to obtain comet assay images. Processes applied during the comet assay experiments are listed below.

- ✓ Blood sample is diluted with phosphate buffered saline (PBS) at the ratio of 1:1.
- ✓ 3 ml Histopaque 1077 is added into 15 ml conical centrifuge tube and temperature is set to room temperature.
- ✓ 3 ml diluted blood is stratified on Histopaque 1077 without mixing to layers in a careful way.
- ✓ It is centrifuged exactly 30 minutes at 400 xg and room temperature. Brake is reset. Acceleration is held at low level.
- ✓ Area between two layers is aspirated when 0,5 cm width of opaque interface is punched with pipet.
- ✓ Opaque interface is added into 15 ml conical centrifuge tube with pasteur pipets.
- ✓ After cells are washed by adding 10 ml DPBS, cells are gently mixed.
- ✓ It is centrifuged throughout 10 minutes at 250 xg.
- ✓ After centrifugation, supernatant is aspirated and discarded.
- ✓ Cell pellets in the tube are diluted with 5 ml DPBS.
- ✓ It is centrifuged throughout 10 minutes at 250 xg.
- ✓ Last 3 steps are repeated and cell pellets are diluted with 0,5 ml DPBS.
- ✓ 1% concentration of normal melting point agarose (NMPA) is prepared with distilled water, it is provided to be completely dissolved by heating in microwave oven.
- ✓ Slides whose one side is frosted are immersed in methanol and are passed flame. Thus, particles and grease remnants on the slides are removed.
- ✓ 1/3 part of frosted area are sinked into 1% concentration of NMPA whose temperature is between 55 and 60 °C. Thus, half of the slides are covered with agarose. This step is called as pre-coating level.
- ✓ It is waited until agarose hardens at room temperature. (minimum 30 minutes)
- ✓ Low melting point agarose (LMPA) solution which is prepared in 1% DPBS is melted in microwave oven.



- ✓ Temperature of LMPA is become stable in water-bath whose temperature is set 37 °C.
- ✓ 30 µl of cell suspension is put into an eppendorf tube. 140 µl from 1% concentration of LMPA at 37 °C is taken and it is mixed with cell suspension.
- ✓ 70 µl mixture which contains LMPA and cell suspension is put on pre-coated slides by means of pipets and the slides are covered with coverslip.
- ✓ The slides are waited in a fridge to make LMPA hard. (5-10 minutes)
- ✓ Before lysis, 1% Triton X-100 and 10% DMSO are added into lysis solution.
- ✓ The slides are put in copling jar which includes cold lysis solution. The slides are waited in the copling jars in a dark room at +4 °C during minimum 1 hour. This stage is called as lysis level. (When lysis time is increased, pH never passes 10,5)
- ✓ The slides are washed three times in full copling jars by using distilled water / DPBS throughout 3x5 minutes at +4 °C and alkaline lysis solution is removed.
- ✓ The slides are put on surface of electrophoresis tank standing on ice. When empty areas stay on surface of the electrophoresis tank, these empty areas are filled with empty slides. Thus, homogeneous current flow is created. Capacity of the used electrophoresis tank during the experiment is 10 slides.
- ✓ Alkaline buffer solution is removed from cold electrophoresis solution until a slim layer is covered all slides. The electrophoresis tank is closed. Bubble creation is prevented.
- ✓ The slides are waited in alkaline electrophoresis buffer for preincubation throughout 40 minutes at +4 °C. Thus, DNA linkages are opened. This stage is called as unwinding level.
- ✓ Electrophoresis process is applied at 300 mA current and 25 V throughout 30 minutes. This stage is called as electrophoresis level.
- ✓ After electrophoresis process finishes, the slides are taken from the electrophoresis tank.
- ✓ The slides are washed in the copling jars three times by means of neutralization buffer. Each washing time is set to 5 minutes for this process. This stage is called as neutralization level.
- ✓ The slides are rinsed with distilled water. Then, the slides are dried throughout 30 minutes at room temperature.

- ✓ After 60  $\mu\text{l}$  ethidium bromide staining solution is added on the slides, the slides are closed with coverslips. This stage is called as staining level.
- ✓ The slides are put on a fluorescent attachment microscope and images are captured by a camera [8-16].

Processes applied during comet assay experiments are illustrated in Figure 2.3.

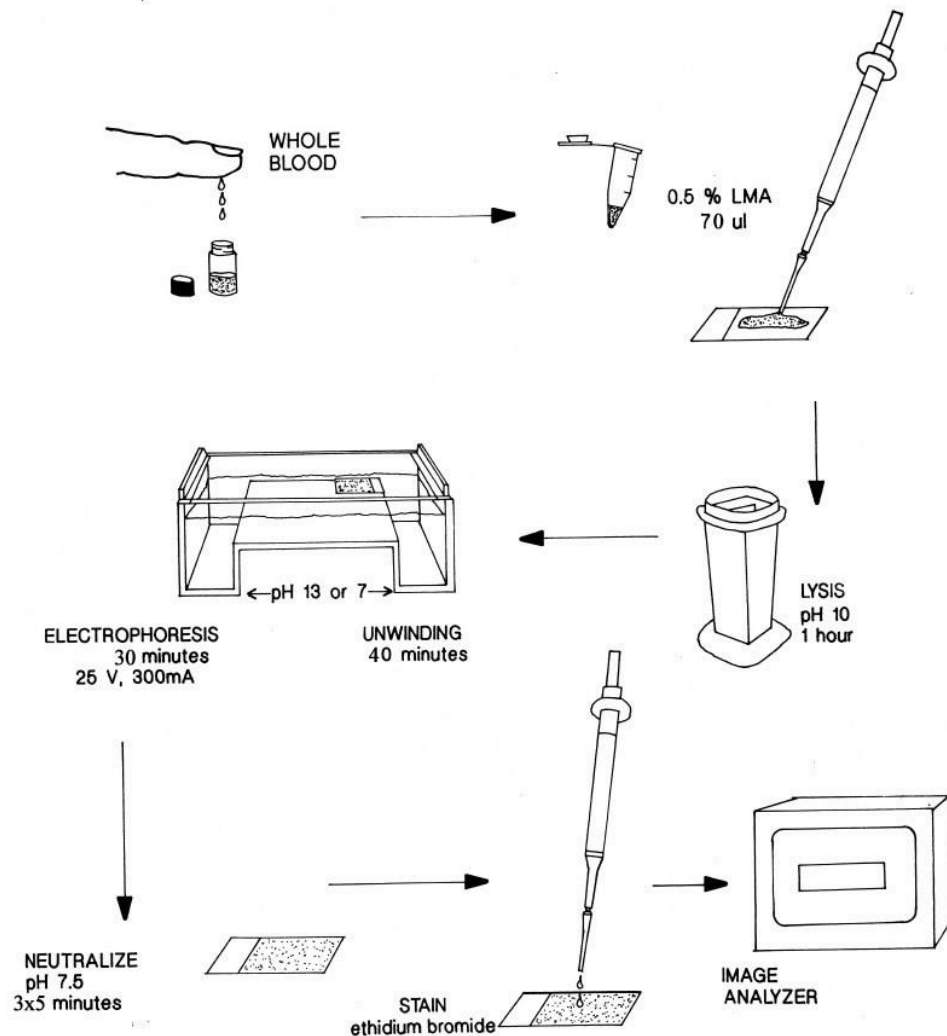


Figure 2.3. Processes of comet assay experiments adapted from [17].

In this thesis study, evaluated parameters related to comet objects are center of head part, length of head part, area of head part, percentage of head part, length of tail part, area of tail part, percentage of tail part and tail moment. A sample of area of head part in Figure 2.4, a sample of center of head part in Figure 2.5, a sample of radius of head part in Figure 2.6, a sample of area of tail part in Figure 2.7 and a sample of length of

tail part in Figure 2.8 are shown. The formulas of these parameters are explained in Chapter 7.

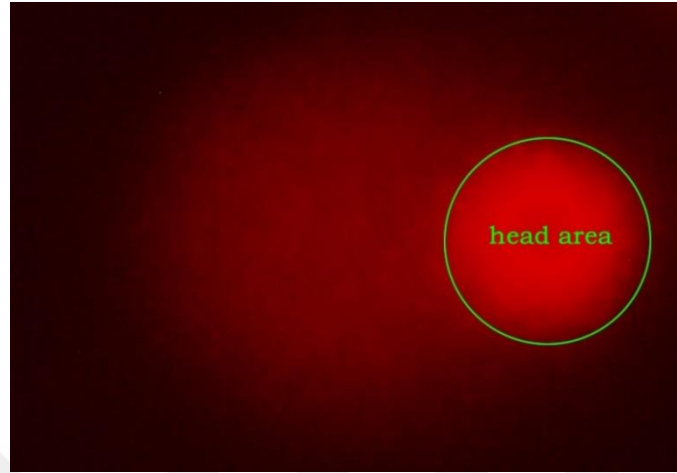


Figure 2.4. Area of head part.

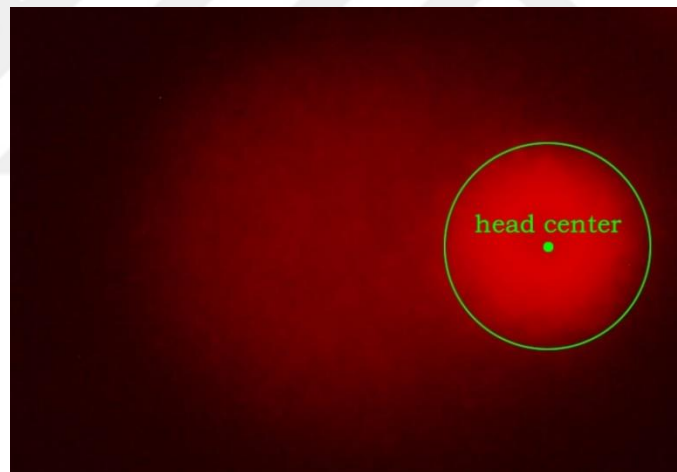


Figure 2.5. Center of head part.

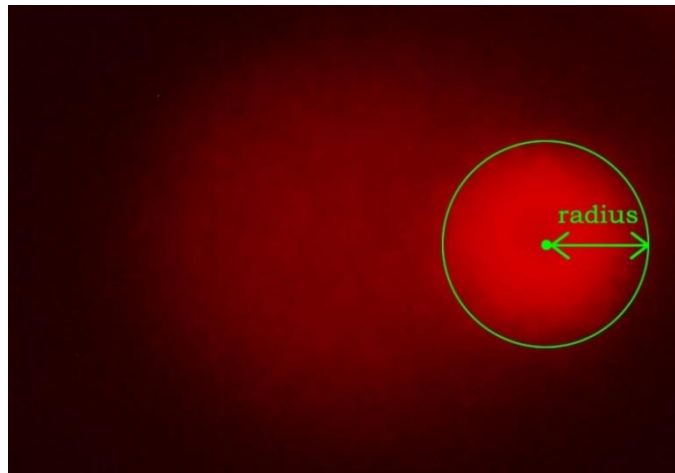


Figure 2.6. Radius of head part.

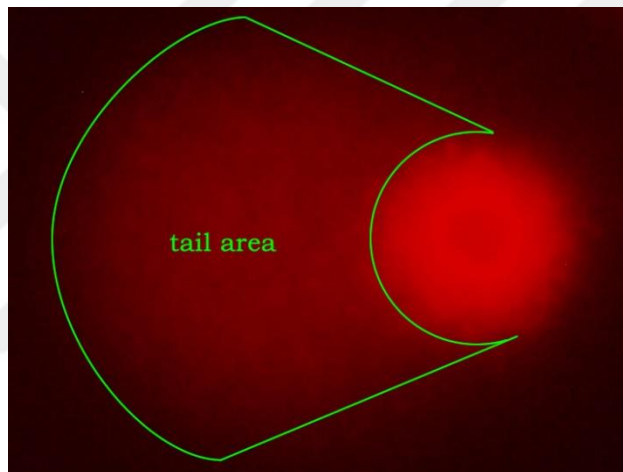


Figure 2.7. Area of tail part.

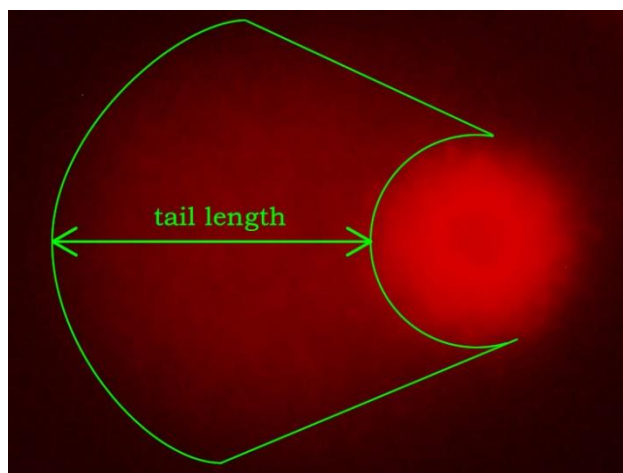


Figure 2.8. Length of tail part.

## CHAPTER 3

### PYTHON AND FUNDAMENTAL PACKAGES

#### 3.1. PYTHON

A short overview of the various aspects of Python programming language and its packages is performed in this chapter. Python is one of the most high-level programming languages. Popularity of Python increases as time progresses. According to TIOBE index for September, Python is in 3<sup>rd</sup> place [18]. Many programming abilities such as scientific computations, desktop application, web application, database programming, microcontroller communication, parallel programming, digital image and signal processing, network programming etc. are provided [19-23]. Python contains various favorable features as:

- ✓ It is free to use.
- ✓ It is available on all most preferred operating systems and platforms such as Windows, Linux or Mac.
- ✓ It is an interpreted programming language. Thus, small parts of codes can be tested on the command line (IDLE part of Python). Compiling or linking are not needed.
- ✓ It has an ability to program faster.
- ✓ It is highly readable and easy to debug.
- ✓ It is easier to write codes syntactically and contains fewer lines of codes than C/C++/Fortran.
- ✓ It consists of many standard packages. Thus, various tasks can easily be performed.
- ✓ Installation of programs written in Python is available on many operating systems and platforms with little or no change.

- ✓ It is a dynamically developed programming language. Thus, declaration of the data type of variables is not a must.
- ✓ It is kept up to date by dedicated developers and user community [19-23].

There exist some unfavorable features in addition to favorable features in Python as:

- ✓ It focuses on the ability to program faster. Thus, the speed of execution is low. A Python program is slower than an equivalent program developed in C. However, it includes fewer line of codes and can easily be programmed to handle multiple data types. This unfavorable feature can be achieved by suitable use of data structures or converting slow running parts of Python codes to C codes.
- ✓ It has an indentation feature that is not optional and makes codes readable. On the other hand, a code including multiple loops and other constructs are indented to the right. This situation makes code readability difficult. As a solution, Python provides some tools like list processing, dictionary. As a result of that, Python sets to reduce this complexity [19-23].

## **3.2. PYTHON FUNDAMENTAL PACKAGES**

Python comes with many built-in packages or libraries. These packages have distinctive features and abilities to fulfil various tasks. Since it is focused on detecting, analyzing and analyzing comet objects on comet assay images, the most utilized packages such as NumPy, SciPy, SciKits, Python Imaging Library (PIL), Matplotlib and OpenCV-Python are shortly explained [19-23].

### **3.2.1. NumPy**

NumPy is a fundamental and powerful package of Python for scientific computation. It presents some abilities such as manipulation with multi-dimensional arrays, sophisticated functions, tools for integrating C/C++ and Fortran codes, useful linear algebra, Fourier transform, and random number capabilities. In addition to them,

NumPy can be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined [19,24,25].

### **3.2.2. SciPy**

SciPy is one of the core packages of Python. It provides effective numerical integration, statistics and optimization routines. Higher order mathematical operations like filtering, image processing, machine learning, statistical analysis can be performed [19,25,26].

### **3.2.3. SciKits**

SciKits is abbreviation of SciPy ToolKits. It is an add-on package for SciPy and developed separately from SciPy. It contains many modules like scikit-image for image processing routines for SciPy, scikit-misc for miscellaneous tools for scientific computing, scikit-learn for machine learning and data mining, scikit-bio for data structures, algorithms and educational resources for bioinformatics, scikit-cmeans for flexible and extensible fuzzy c-means clustering, scikit-fuzzy for fuzzy logic toolkit for SciPy etc. [19,27].

### **3.2.4. Python Imaging Library (PIL)**

PIL is a core image library of Python. It provides image processing abilities to Python interpreter. Operations like image reading, image writing, point operations, filtering with a set of built-in convolution kernels, color space conversion, resizing, rotation, histogram can be performed on any format of images [19,28].

### **3.2.5. Matplotlib**

Matplotlib is Python 2D plotting library that provides functions to plot and visualize other forms. Graphics, images, histograms can be shown by matplotlib. Some features like giving graph title, axis title, changing colors, legend, axis intervals etc. can be managed [19,29].

### **3.2.6. OpenCV-Python**

OpenCV is a library developed for various programming languages such as Python, Java, C, C++ etc. In addition to that, it is available on many operating systems or platforms. OpenCV-Python is a library developed for Python. It provides some features such as managing GUI features, performing core operations, image processing, object detection, video analysis, camera calibration, machine learning and computational photography [30].





## CHAPTER 4

### LITERATURE REVIEW

Recently, there is a growing popularity in terms of computer hardware and software in biomedical area. Medical imaging and automated analysis systems which make popularity grow are two of a lot of topics about biomedical area. While medical imaging presents visual information about images taken from patients, automated analysis systems detect abnormal features on images and present numerical results in a fast manner about diseases and progression of diseases. Thus, medical imaging and automated analysis systems confront with users as tools that make their lives easier.

Digital image processing or image processing is one of the most popular topics in computer science. Digital image processing is accepted as a subtopic of digital signal processing. Digital image processing is interested in digital images. The input and output of systems are digital images. There are many algorithms about image processing like detecting objects, correcting images, extracting information from images, classification, filtering, segmentation, and performing analysis on images etc. After performing image processing algorithms on input images, the output of the systems generally becomes a digital image, graphics and numerical results.

Automated analysis of DNA damage on comet assay images is one of the unusually studied topics in biomedical area. There are some critical points that may lead to wrong interpretations such as variability of images in terms of color or gray level, the morphology of comet objects, high number of comet objects on images, heaviness of the DNA damage, elimination of artefacts and blurry objects created by environmental factors and overlapped objects created by DNA.

In the literature, there have been few studies that use image processing algorithms about automated detection of DNA damage on images. The studies have concentrated

on dividing comet structure into two parts such as head part and tail part. It has been decided whether DNA have damage or not by calculating head intensity, head area, head length, tail intensity, tail area, tail length and tail moment.

Number of publications about analysis and quantification of DNA damage in computer science field versus year till 2018 is presented in Table 4.1.

Table 4.1. Number of publications versus year.

<b>Year</b>	<b>Number of Publications</b>
2000 and before	5
2001	1
2002	1
2003	1
2004	1
2007	1
2008	1
2009	2
2012	2
2013	1
2014	2
2015	2
2016	2
2017	4
2018	-
<b>Total</b>	<b>26</b>

In this chapter, researches focused on semi and fully automated analysis of comet assay images are described.

Gyori et al. have developed an open-source software application named as OpenComet. In this study, after preprocessing stage which is used to remove noises, adaptive thresholding method has been applied using intensity histogram to obtain binary images whose black pixels show background and white pixels show comets. Because each comet region has had different brightness, adaptive thresholding method has been chosen to obtain each comet region. Convex and symmetric shape properties have been utilized to recognize overlaps and irregular shapes. When these regions have been extracted from images, the remaining regions have been accepted as comets. To find comet head and tail, brightest regions have been found because it has been assumed that brightest regions have been almost comet heads. To separate both, intensity profile analysis has been applied [2]. Sensitivity has been calculated as 63.95% [31].

Sreelatha et al. have proposed an effective and fully automatic method to detect comets on very noisy silver stained comet assay images. Their software program has measured comet length, tail length, head diameter, percentage DNA in head, percentage DNA in tail and tail moment. The analysis has been divided into three stages such as comet detection, comet segmentation and comet quantification. Shading correction has been used to correct images. Contrast enhancement has been applied to discriminate between comets and background. Gaussian filtering has been applied to smooth images and get elongated blobs. To entirely discriminate them, second contrast enhancement has also been applied by taking the average of the intensities over four square areas specified at the four corner points of images. Then, images have been applied thresholding to obtain binary images by using Otsu's thresholding method. To remove the eight connected low intensity objects existing at image borders, the image complement has been taken. Furthermore, smaller artefacts in the background have been removed by using binary morphological opening operation. The contours of detected objects have been smoothed by using morphological closing operation. Overlapped comets have been removed by thresholding each of the individually cropped objects at 80% of the maximum intensity. At that point, images have had comets, the gel or air artefacts. The gel and air artefacts have been removed by analyzing the comet profiles. Sensitivity has been calculated as 89.30% [32].

Sreelatha et al. have proposed an algorithm to improve automatic detection of true comets in their studies. They have divided their studies into three stages such as comet identification, comet segmentation and comet quantification. Shading correction, homomorphic filtering, Otsu's thresholding method, morphological filtering, and morphological thickening have been applied on images to detect comets. Sensitivity has been calculated as 93.17% [31].

Smolka and Lukac have used three different methods to extract head and tail from the images in their studies such as probabilistic approach, region based segmentation and active contour segmentation. Although these methods have obtained different results, all of them have been similar to assessments of human observer [4].

Sansone et al. have proposed an automated comet analysis algorithm in their studies. The study has been divided into two steps such as comet detection and comet segmentation. In the first step, comet detection has been realized by using the Gaussian pre-filtering and morphological operators. In the second step, comet segmentation has been performed by using the fuzzy clustering method. High sensitivity results have been obtained for both steps in their studies [33]. Sensitivity has been calculated as 78.96% [31].

Böcker et al. have developed an automated analysis system based on self-developed software and hardware for DNA damage in their studies. The system has also needed human interaction. The analysis has been divided into two parts such as automated cell recognition with comet classification and comet quantification. Algorithms based on mathematical morphology have been used in preprocessing stage, segmentation stage and feature classification stage. They have reported that histogram analysis, entropy maximization, k-means clustering, and contour based procedures have failed in their studies. Hence, two procedures have been used to calculate threshold value and make image content analysis. Adaptive threshold method has been used to improve segmentation results. Shading correction, morphological opening and closing have been used to enhance image quality. Morphological thickening has been used to obtain head region of comet. To obtain results faster, parallel programming has been used in

their self-developed software. Sensitivity has been calculated as 95.20% and Specificity has been calculated as 92.70% [34].

Vojnovic et al. have applied a median filter whose window size is 21x21 and performed image normalization at the preprocessing stage. Median filter has been applied on raw images. Black/White level correction has been performed to normalize images. After that, region of interest (ROI) of each comet have been obtained by using thresholding on normalized images. The thresholding value has been set as the intensity value corresponding to 20% of the peak frequency of the histogram. Then, each individual comet has been separated by using binary region growing. When one of five overlapping criteria has occurred, overlapped comets have been rejected. After that, the average intensity value of edge of rectangular ROI has been calculated and then, subtracted from ROI thumbnail image to make ROI homogeneous. When unsuccessful result images have been obtained, two-dimensional quadratic function with the use of a general least square algorithm has been applied to correct original images of these unsuccessful result images. After that, Compact Hough and Radial Map (CHARM) algorithm with Sobel edge detection filters has been used to find where center positions of bright circular objects have been. When center points have been found, object bounds have been obtained by performing radial searching based on the response of the Sobel filters. Therefore, head detection has been performed. Tail length has been calculated by obtaining the distance between the center point of the comet head and the last non-zero pixel of comet profile [35].

Konca et al. have developed a public domain program called as CASP under GNU License for detection of comets. The head center, the head radius, head area, tail length and tail area parameters have been calculated by their algorithms. The program can detect comets oriented from the left-hand side and right-hand side. The program has calculated maximum intensity value and minimum background value by looking at all pixel intensities. After that, users can set a threshold value between maximum and minimum intensity values for thresholding. To find head center, the program has looked at 80-100% of the maximum intensity value to obtain brightest points. By finding center of mass between all brightest points, head center has been detected. Then, the program has scanned leftwards points until it has found the point below the

threshold value. When the program has found the leftmost point, the program has scanned up and down by starting from the line between head center and leftmost points. As a result, all pixels on the left edges of head have been found. After calculating distance between head center and all edge points, mean value of distances has been accepted as head radius value. To find tail area, two methods have been implemented. In the first method, an area has been found at the right side of the head by the compact region adjacency. In the second method, pixels which exceeds threshold value have been accepted as tail area points. Distance between the rightmost point of tail and rightmost point of head has been calculated. Thus, tail length has been found [36].

Rivest et al. have presented a semi-automated method to detect primary DNA damage. Four comet images have been used in this study. After users have manually created markers for comets and background, the system has roughly understood the locations and contours of each comet and background. Then, images have been filtered by using a morphological closing with a vertical structuring element whose size is 11 pixels long. After that, another closing with 5x5 square has been applied. Therefore, small dark debris have been removed on images. Finally, progressive refinements based on watershed transformation has been applied to obtain final edges of comets. Then, they have measured comet surface, tail length and average intensity [37].

Helma and Uhl have developed a publicly available image-analysis system for U.S. National Institutes of Health (NIH) images by writing in Pascal-like macro language. While the macro has determined a threshold value automatically to separate comets with background, the macro has allowed to set a threshold value manually for users. Thus, borders of comets have been determined in this way. By using wand tool, users have selected individual cells. The calculation of the area and DNA intensity has been done by using functions of NIH images and copying the selected cells to another window. As a result of this, location of the head has been found. By looking at maximum intensity values, center of head has been determined. Difference between center of head and leftmost edge with high intensity has given to radius of head. Difference between rightmost edge of head and right-most edge of border of comet has given the tail length value [38].

Böcker et al. have published a technical report for semi-automatic image analysis of measurements. Both a black image captured with closed light excitation shutter and a white image obtained from a uniformly fluorescent field have been taken experimentally. Image shading correction has been applied to eliminate noises on captured images by using both image types. After that, image quality has been enhanced with high values in grey level intensity for white image and lower values for black image. After preprocessing stage, selection of cell, image segmentation and cell-feature quantification have been applied. At the first step, a fixed circle has been located on images. By moving microscope stages, the circle has been attached with comet head. Then, the diameter of the circle has been adjusted according to comet head. To distinguish comet from background, images have been converted to binary image with an adjustable threshold value. Image segmentation has done by using contrast stretching with look-up table and median filter. Images have been stretched to eight-bit range. Thereafter, images have been applied five openings and closings to remove small objects and gaps. As a result, comet parameters have been evaluated and it has taken less than 2 seconds per one comet [39].

Harrison et al. have written a program with IDL programming language. They have firstly generated binary images by using a dynamic threshold value. After that, closing operation with 4x4 disc structure has been applied. Images have been smoothed by average or box filter. Then, opening operation with 4x4 disc structure has been applied. Images have been rotated into a standard and horizontal orientation. For further steps, binary rotated images have been used to reduce background zero intensity values. First order moments have been used to find where comet centroid has been. An angle value has been calculated by using second order moments to find the angle between comets and axis. After finishing these steps, comet moment ratio and tail shape ratio have been calculated as well [40].

Gonzalez et al. have developed a software program named as CellProfiler to automatically identify, quantify and export comet assay information. The mixture of Gaussian method and expectation-maximization algorithm have been used to identify comets. The used methods have not been explained in the paper. Mainly, performance

of CellProfiler software program has been compared with performance of CASP software program in this paper [41].

Sreelatha et al. have developed an automated tool named as CometQ for the detection and quantification of DNA damage using comet assay images. Their comet assay analysis has been separated into four stages as classifier, comet segmentation, comet partitioning and comet quantification. In the classifier stage, support vector machine (SVM) algorithm has mainly been performed to classify type of images as silver stained images or fluorescent stained images and classify type of comets as lightly, moderately or heavily damaged on related type of images. In comet segmentation stage, four different segmentation methods have been developed based on each type of image. The major techniques in these segmentation methods are shading removal, image enhancement, thresholding, noise removal, and detection of actual comets. In comet partitioning stage, actual comets are classified as head clustering, halo clustering and tail clustering with morphological operations. In comet quantification stage, comet parameters have been calculated. More than 600 images have been used in this study. Positive predictive value is calculated as 90.26% and sensitivity is calculated as 93.34 [42].

Lee et al. have developed a method to classify DNA damage patterns on comet assay images in their studies. The method has consisted of three stages as detection, adjustment and analysis. In detection stage, RGB images have been converted to grayscale images and applied preprocessing methods based on scale bar correction, median filter and moving average filter. Pixel intensities have been classified by K-means clustering algorithm as 0 (background) and 1 (comet). Then, adjacent pixels have been handled as pixels of same objects using membership matrix. In adjustment stage, objects whose pixels locate on either first row, last row, first column or last column of the matrix have been removed as boundary objects. A method based on thresholding distance has been performed to remove apoptosis cells. Canny edge detector algorithm has been used to eliminate overlapped comets. In analysis stage, comet parameters have been calculated. The average classification accuracy has been calculated as 86.80% for 20 test data sets including more than 300 images [43].



Mani and Manickam have developed a standalone tool named as CoMat using Matlab and Visual Basic for the detection and quantification of DNA damage. While Visual Basic has been performed for the graphical user interface (GUI), Matlab has been performed for analysis and calculations [44].

Comparison of publications according to sensitivity parameter is presented in Table 4.2. Image type, application type and the used methods to analyze DNA damages and calculate comet parameters on comet assay images are presented in Table 4.3.

Table 4.2. Comparison of publications according to sensitivity.

<b>Reference</b>	<b>Year</b>	<b>Sensitivity (%)</b>
Böcker et al. [34]	1999	95.20
Sansone et al. [33]	2012	78.96
Gyori et al. [2]	2014	63.95
Sreelatha et al. [32]	2014	89.30
Sreelatha et al. [31]	2015	93.17
Sreelatha et al. [42]	2016	93.34

Table 4.3. Image type, application type and the used methods.

<b>Reference</b>	<b>Image Type</b>	<b>Application Type</b>	<b>Comet Detecting Methods</b>
Gyori et al. [2]	Grayscale	Fully-automated	Adaptive thresholding, region shape filter, intensity profile analysis
Smolka and Lukac [4]	Original Grayscale	Unknown	Probabilistic approach, region based segmentation, active contour segmentation
Sreelatha et al. [31]	Original	Fully-automated	Gradient calculation, thresholding, rejection of ROIs, local texture-based fuzzy, fuzzy c-means clustering

Sreelatha et al. [32]	Original	Fully-automated	Shading correction, contrast enhancement, gaussian filtering, thresholding, elimination
Sansone et al. [33]	Original	Fully-automated	Gaussian pre-filtering, mathematical morphology, fuzzy c-means clustering
Böcker et al. [34]	Grayscale	Semi-automated	Shading correction, mathematical morphology
Vojnovic et al. [35]	Grayscale	Fully-automated	Thresholding, Compact Hough and Radial Map
Konca et al. [36]	Colorful or Grayscale	Semi-automated	Intensity profile analysis
Rivest et al. [37]	Grayscale	Fully-automated	Mathematical morphology, watershed transformation
Helma and Uhl [38]	Grayscale	Semi-automated	Intensity profile analysis, thresholding
Böcker et al. [39]	Grayscale	Semi-automated	Shading correction, thresholding, intensity profile analysis
Harrison et al. [40]	Binary	Unknown	Thresholding, mathematical morphology, first order moment, second order moment
Gonzalez et al. [41]	Binary	Fully-automated	Thresholding, mixture of Gaussian method, expectation-maximization algorithm
Sreelatha et al. [42]	Grayscale	Fully-automated	Shading removal, enhancement, SVM, thresholding, morphological operations, profile analysis
Lee et al. [43]	Grayscale	Fully-automated	Removal of scale bar, K-means clustering, Canny edge detector
Mani and Manickam [44]	Grayscale	Fully-automated	Connected component labeling

## CHAPTER 5

### UNIFIED MODELLING LANGUAGE DIAGRAM OF THE DEVELOPED APPLICATION

With the gain of popularity on usage of object oriented language, demand for formal, independent from programming languages and standard modelling language has been increased dramatically [66]. Object Modelling Technique (OMT) developed by James Rumbaugh et al., Object Oriented Analysis & Design (OOAD) developed by Grady Booch and Object Oriented Software Engineering (OOSE) developed by Ivar Jacobson were combined. As a result of this, one single modelling language called as UML was formed. Notations related to these three studies exist in UML [45-48].

UML diagram is a representation of components or elements of a system including a number of diagrams that present what attributes, methods, connections are required in a system, how they communicate with each other, how many actors use a system and what features are used by actors [45,49].

The first condition to develop a software professionally is to do overall requirement analysis and logical design before coding, even if there are claims about the fact that analysis and design stages are time loss. More time is lost to overcome problems during process of coding which starts without doing any pre-study.

In UML, there exist many different diagrams which document modelling created by approaching from different point of views.

- ✓ Behavioral Modelling
  - ✓ Activity Diagram
  - ✓ State Diagram
  - ✓ Requirement Diagram

- ✓ Sequence Diagram
- ✓ Communication Diagram
- ✓ Timing Diagram (For real time systems)
- ✓ Structural Modelling
  - ✓ Class Diagram
  - ✓ Object Diagram
  - ✓ Deployment Diagram
  - ✓ Composite Structure Diagram
  - ✓ Component Diagram
- ✓ Functional Modelling
  - ✓ Use Case Diagram

Behavioral modelling or behavioral diagram depicts behaviors of a system. Structural modelling or structural diagram depicts structure of a system. It focuses on structural features of a system or function. Functional modelling depicts that a system can perform what jobs and how actors can use a system with use case diagrams [45].

## **5.1. USE CASE DIAGRAM**

Use case diagram is the first stage of object oriented design. Use case diagram depicts use cases, actors and communications between use cases and actors in a system. In the analysis stage, it is used with activity diagrams. The main goal of using case diagram is to clarify what kind of functional requirements a system need and what a system presents for users. Use case diagram does not present any technical requirements [45,49]. Use case diagram of the system in this thesis study is shown in Figure 5.1.

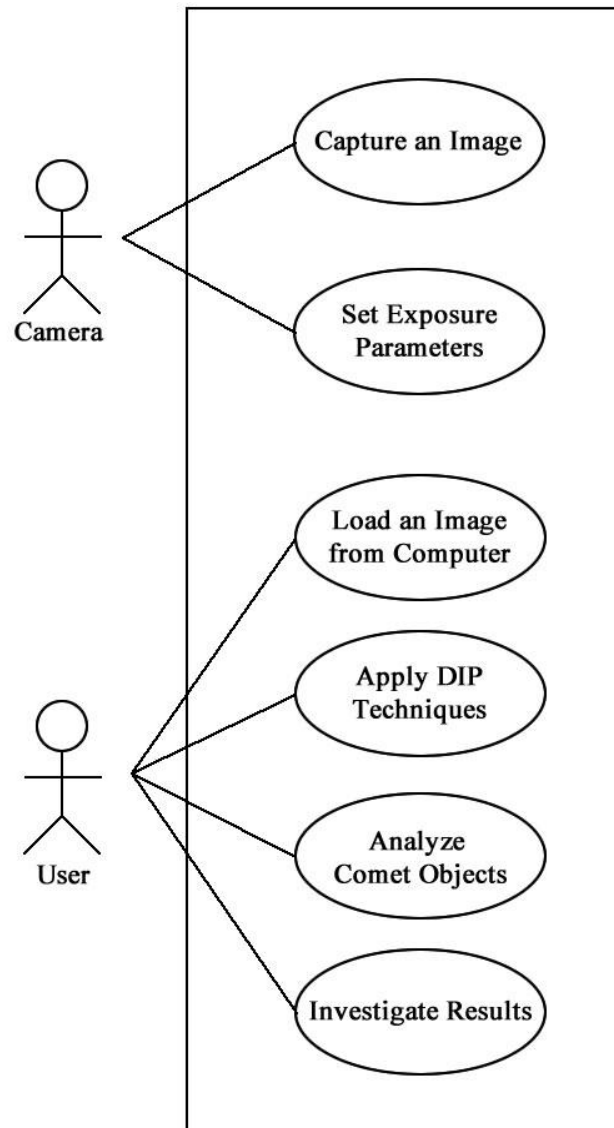


Figure 5.1. Use case diagram.

### 5.1.1. Use Case 1

User can load images by using either connected camera or computer's memory. If user selects connected camera, user can capture images with either automatic or manual exposure settings. If user selects computer's memory, user loads images from any location on computer. User can analyze loaded images directly or after using digital image processing techniques manually. After analysis, user can investigate comet objects, drawn head and tail objects, head length, tail length, head area, tail area, head percentage, tail percentage, tail moment and grade level.

### **5.1.2. Use Case 2**

If user selects connected camera to load images, user can capture images by using either automatic exposure or manual capture. If user selects automatic exposure, user can capture images without setting any parameters. If user selects manual exposure, user can set parameters like refresh rate, minute, second, millisecond, gain, contrast and gamma.

### **5.1.3. Use Case 3**

User can manually apply digital image processing techniques like rotating right, rotating vertically, zooming in, zooming out, logical and, logical or, logical xor, erosion, dilation, opening, closing, arithmetic addition, arithmetic subtraction, arithmetic multiplication, arithmetic division, inversion, median filter, mean filter, min filter, max filter, gaussian filter, histogram stretching, histogram equalization, contrast control and brightness control. User can undo and redo processes. After finishing applying digital image processing techniques, user can analyze loaded images.

## **5.2. RELATIONSHIPS BETWEEN USE CASES**

Use cases may have relationships with each other in addition that use cases have relationships with actors. Relationship between two use cases stands for dependency. It is indicated with dashed lines. Relationship between an actor and a use case stands for association. It is indicated with solid lines. Name of a relationship is indicated with a stereotype. Stereotype gives information about relationships. Three main stereotype names are used in UML such as <<uses>>, <<extends>> and <<includes>>. When one use case uses a behavior of another use case to perform an interaction, <<uses>> stereotype is used as a relationship name. Use connection is used to model how use cases used by actors are performed. When one use case is a variation of another use case (like inheritance or polymorphism), <<extends>> stereotype is used as a relationship name. When one use case includes behavior of another use case, <<includes>> stereotype is used as a relationship name. Include connection is used to avoid adding a method into many use cases. Use case performs a process according to

result provided by included use case [45,49]. Relationships between use cases of the system in this thesis study are shown in Figure 5.2.

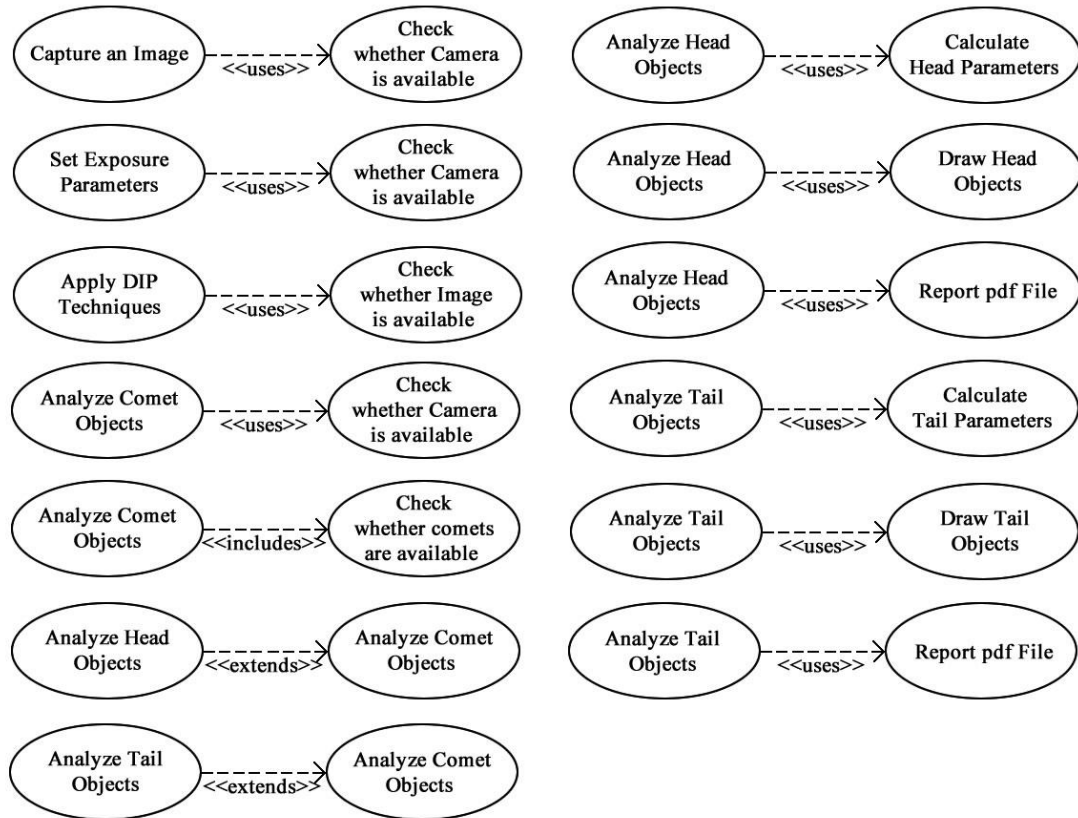


Figure 5.2. Relationships between use cases.

### 5.3. ACTIVITY DIAGRAM

Activity diagram is used to illustrate workflow of a system. Activity diagram is a complement of use case diagram since use case diagram presents only what a system can perform processes. However, activity diagram presents which stages and how processes are passed. Activity diagram is presented after use case diagrams are illustrated [45,49].

Actions located in workflow of a system compose of a complete activity. The main process performed in activity diagram is to present transitions among actions located in workflow of a system [45,49]. Activity diagrams of the system in this thesis study are shown in Figure 5.3-5.4.

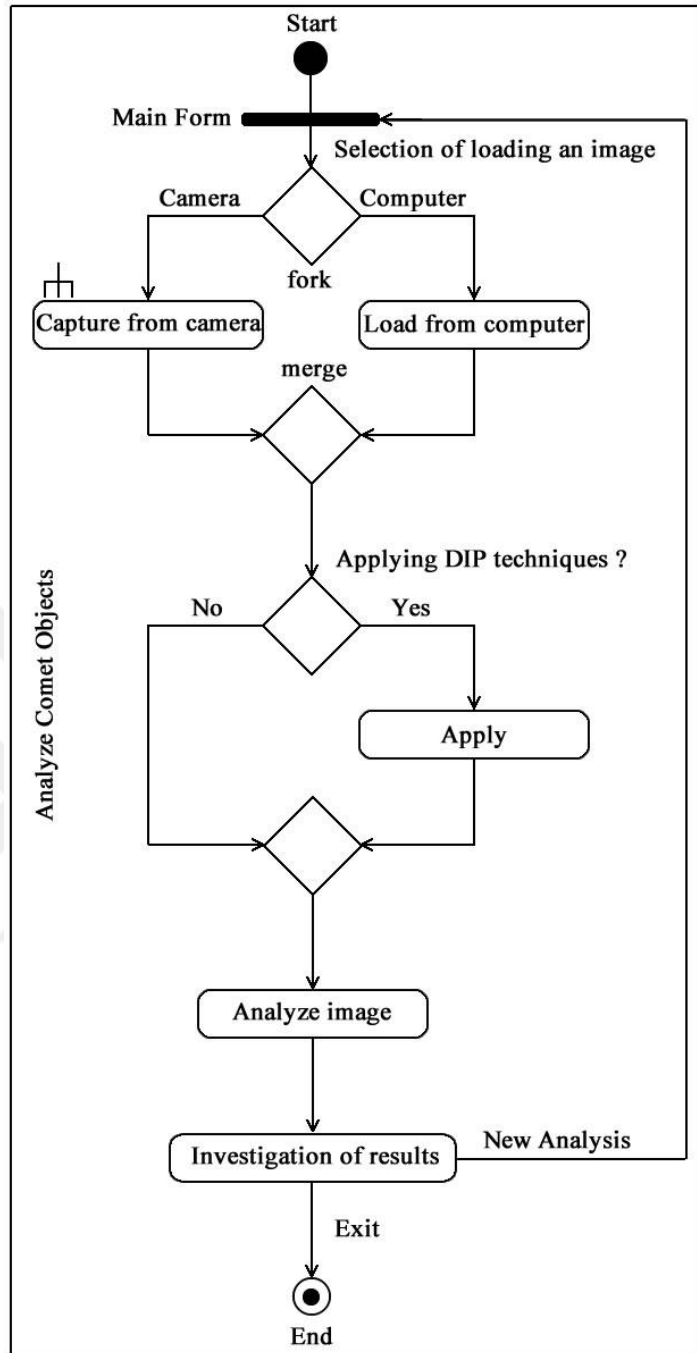


Figure 5.3. Activity diagram of analyzing comet objects.



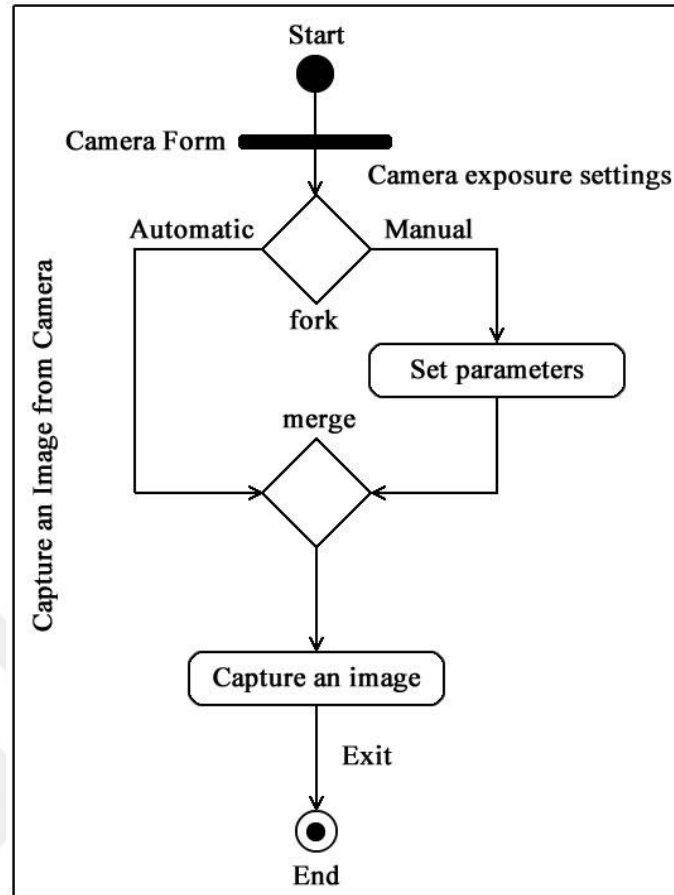


Figure 5.4. Activity diagram of capturing an image from camera.

#### 5.4. REQUIREMENT DIAGRAM

Requirement diagram is used for requirement management and documentation. Requirement management is not developing a requirement. Requirement management is to model, update and review requirements. Thereafter, requirements are written down documents named as system requirement specification (SRS) by system users like analysts, programmer, customer, tester and so on. Documents play an important role for conceptual analysis process. Moreover, documents serve as project concept documents [45,49]. Requirement diagram of the system in this thesis study is shown in Figure 5.5.

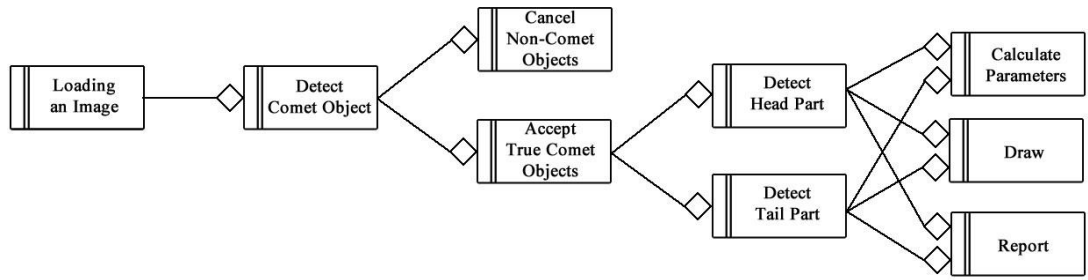


Figure 5.5. Requirement diagram.

## 5.5. COMMUNICATION DIAGRAM

Communication diagram is a diagram that aims at modeling many instances working to perform a scenario. Messages between instances in communication diagram can be either synchronous or asynchronous. Each message is given a number starting from one and increasing by one. Hence, order of messages between instances is specified by numbers.

Sequence diagram is similar to communication diagram. Communication between instances is provided with time concept. Hence, order of messages between instances is specified by time [45,49]. Communication diagrams of the system in this thesis study are shown in Figure 5.6-5.8.

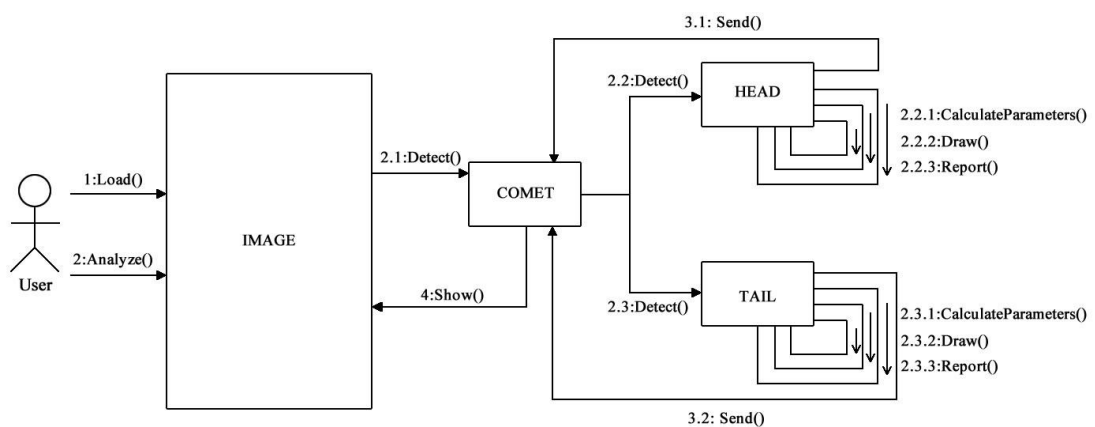


Figure 5.6. Communication diagram of analyzing an image.

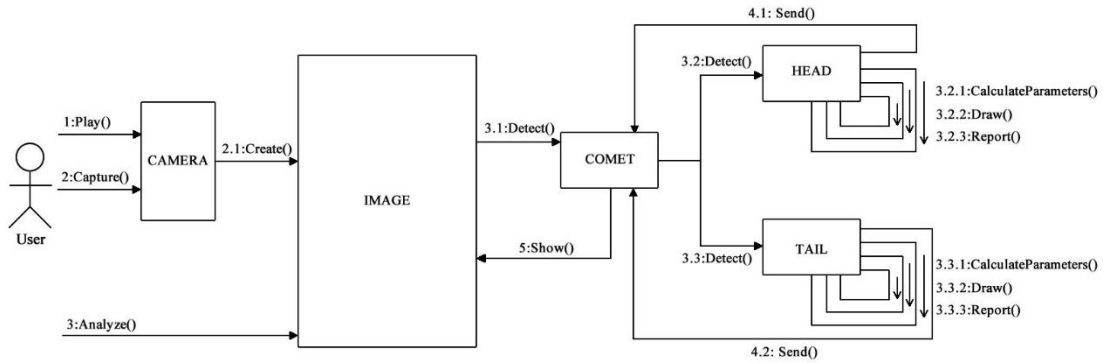


Figure 5.7. Communication diagram of capturing and analyzing an image.

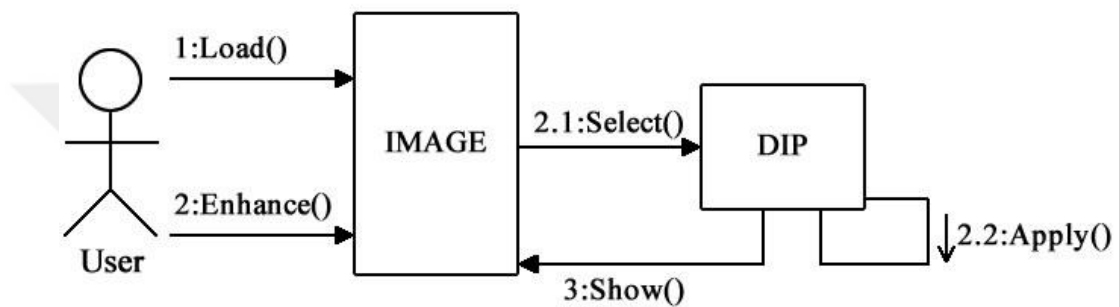


Figure 5.8. Communication diagram of applying image processing techniques.

## 5.6. RELATIONSHIPS BETWEEN OBJECTS

Relationships between instances representing classes are defined as Association, Aggregation, Composition, Generalization and Specification, Dependency, Usage and Realization [45,49].

### 5.6.1. Association

Association is the most basic and frequently encountered relationship type between instances. In Association relationship, a class programmatically carries an instance reference from another class. An example of Association relationship type is that connection property of SqlCommand class carries an instance reference of SqlConnection class. An arrow indicates related objects [45,49]. The representation of Association relationship is shown in Figure 5.9.

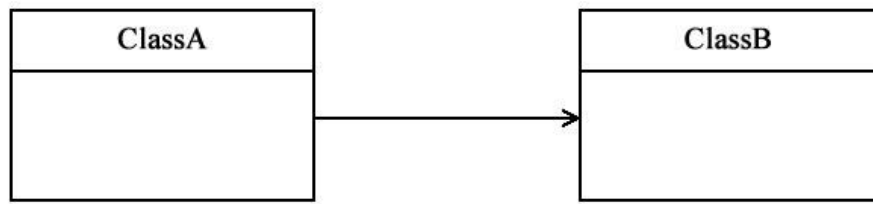


Figure 5.9. The representation of Association relationship.

### 5.6.2. Aggregation

Aggregation is a special type of Association. Aggregation is used where an instance of class includes a reference of an instance of another class. But, references of an instance of another class can be included in instances of other classes. A diamond mark is used in representation of Aggregation relationship [45,49]. The representation of Aggregation relationship is shown in Figure 5.10.

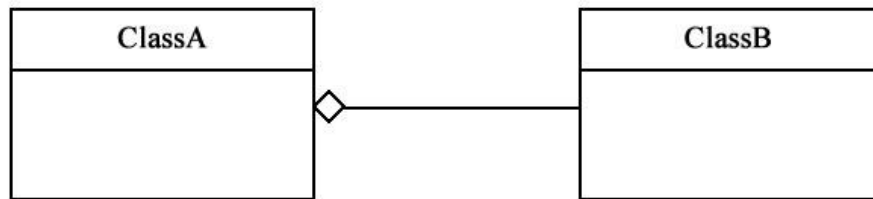


Figure 5.10. The representation of Aggregation relationship.

### 5.6.3. Composition

Composition is a special type of association. Composition is used where an instance of class includes a reference of an instance of another class. But, references of an instance of another class cannot be included in instances of other classes. This relationship can be described as organic dependence of two objects. A full diamond mark is used in representation of Composition [45,49]. The representation of Composition relationship is shown in Figure 5.11.

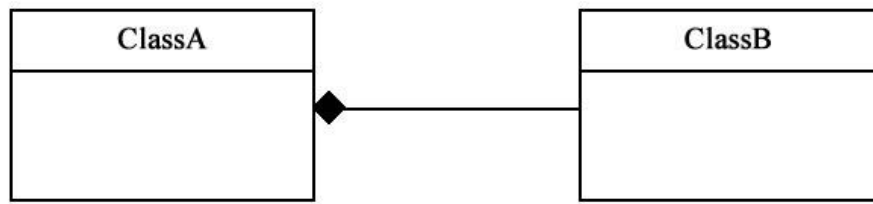


Figure 5.11. The representation of Composition relationship.

#### 5.6.4. Generalization and Specialization

Generalization and Specification relationship is defined as a relationship between base classes and subclasses [45,49]. The representation of Generalization and Specialization relationship is shown in Figure 5.12.

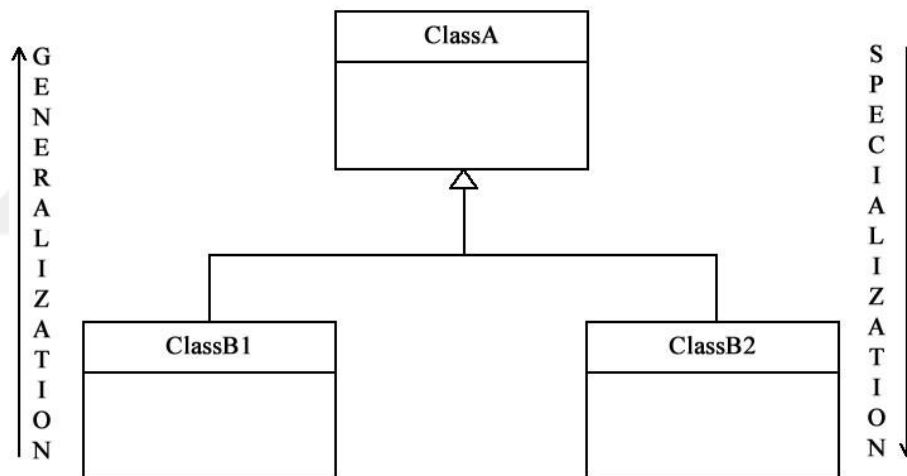


Figure 5.12. The representation of Generalization and Specification relationship.

Relationship among classes of applications is called as object diagram that comprises of Association, Aggregation and Composition relationships. Object diagram of the system in this thesis study is shown in Figure 5.13.

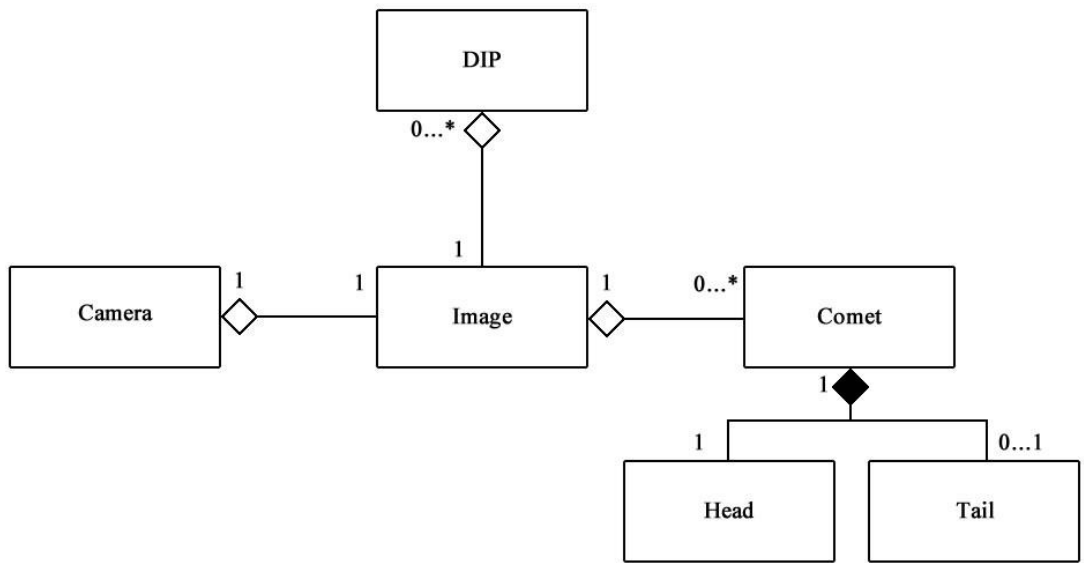


Figure 5.13. Object diagram.

## CHAPTER 6

### MATERIALS

Materials used during this thesis study are introduced in this chapter. Comet assay experiments have been performed to obtain comet assay images throughout this thesis study. Chemical materials are listed in Table 6.1 to perform comet assay experiments.

Table 6.1. Comet assay materials.

<b>Material Name</b>
EDTA
Histopaque-1077
DPBS
Triton X
DMSO
NaCl
HCl
Trisma base
NaOH
LMPA
NMPA
Distilled water
Ethidium bromide

Since cells are located inside preparates, a microscope and a camera are needed to monitor and make them larger. Olympus CX 31 trinocular microscope with fluorescent attachment used during comet assay experiments is shown in Figure 6.1.



Figure 6.1. Olympus CX31 trinocular microscope.

Olympus E-330 pro camera standing on the microscope is utilized to capture comet assay images. After connection between the camera and the computer is established, images are transferred from memory of the camera to computer. Images are captured as Tagged Image File Format (TIFF). 40X ocular objective lens is preferred. Properties of the camera are listed in Table 6.2.

Table 6.2. Properties of Olympus E-330 pro camera.

<b>Property</b>	<b>Value</b>
Image dimensions	3136x2353 pixels
Resolution	314 dpi
Bit depth	24
Color representation	sRGB
F-stop	f/0
Exposure time	¼ sec.
ISO speed	ISO-400



Properties of the used computer throughout this thesis study are listed in Table 6.3.

Table 6.3. Properties of the used computer.

<b>Product</b>	<b>Property</b>
Central processing unit (CPU)	Core i7 4.00 GHz
Read access memory (RAM)	4x8 GB DDR3 1600 MHz
Hard disc	256 GB
Cache memory	8 MB
Display card	6 GB 384 bit
Monitor	22 inches Full HD
Operating system	Windows 10 (64-bit)

In this thesis study, 2476 comet assay images have been captured and analyzed. Images were evaluated by a clinical expert. All individual candidate comet objects were manually classified as either blurry comet or non-blurry comet. All individual candidate comet objects were manually classified as either non-overlapped comet or overlapped comet. All true comet objects were manually graded. Confusion matrix that is very frequently used tool to summarize and compare experimental results with clinical experts' results is used for validation. Therefore, sensitivity, specificity and accuracy corresponding to elimination of blurry objects, elimination of overlapped comets and grading damage level are calculated and compared with the clinical expert's evaluation.

## CHAPTER 7

### METHODS

The fully automated application focuses on analysis and quantification of comet assay images under the processes of elimination of non-comet objects, grading damage level, calculating comet parameters and presenting results. The flow chart of the algorithm for analysis and quantification of comet assay images is shown in Figure 7.1.

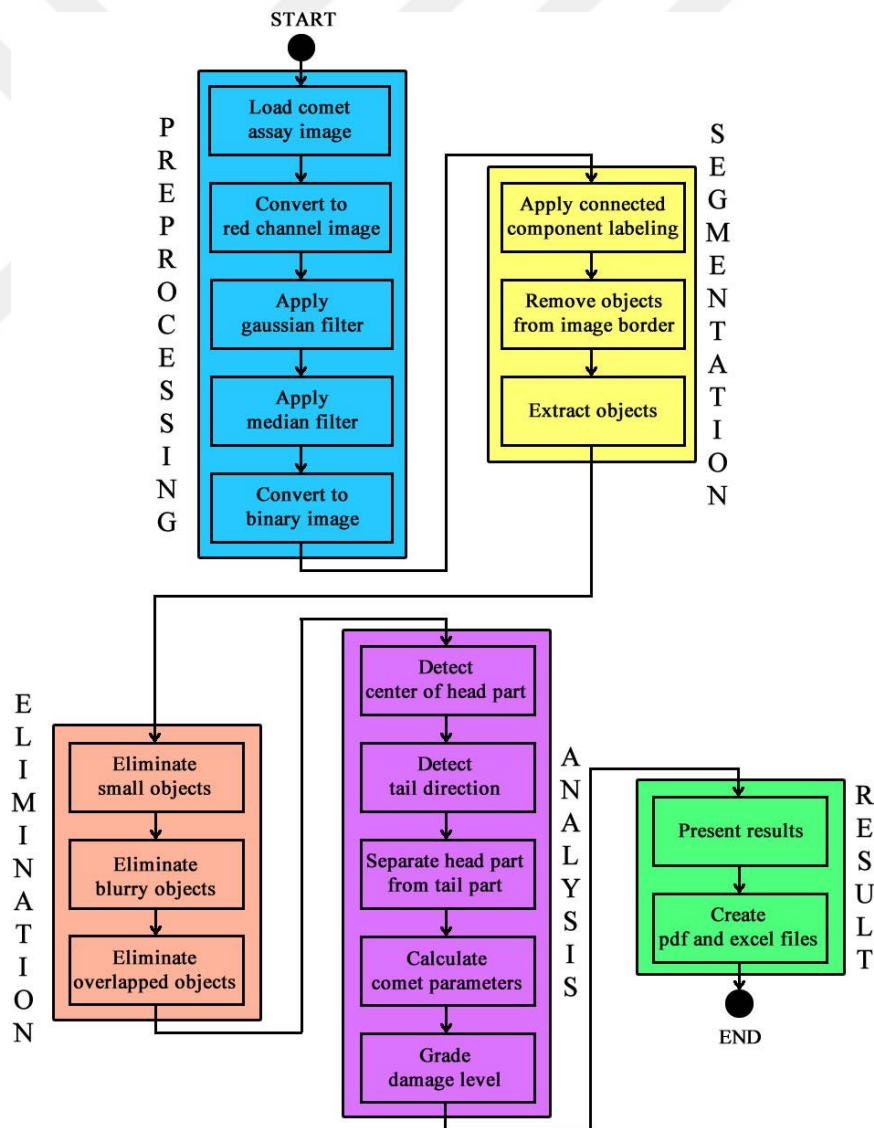


Figure 7.1. The flow chart of the algorithm.

## 7.1. PREPROCESSING STAGE

In this stage, the main aim is to smooth and enhance comet assay images and then convert to binary images before performing analysis. First of all, a comet assay image is loaded from either capturing a live image from the camera or any location on computer. When a comet assay image is loaded, a folder with image name is created in application folder. The loaded image is converted to red channel image since there is no intensity on green and blue channels. The histograms of red, green and blue channels of a sample comet assay image are shown in Figure 7.2.

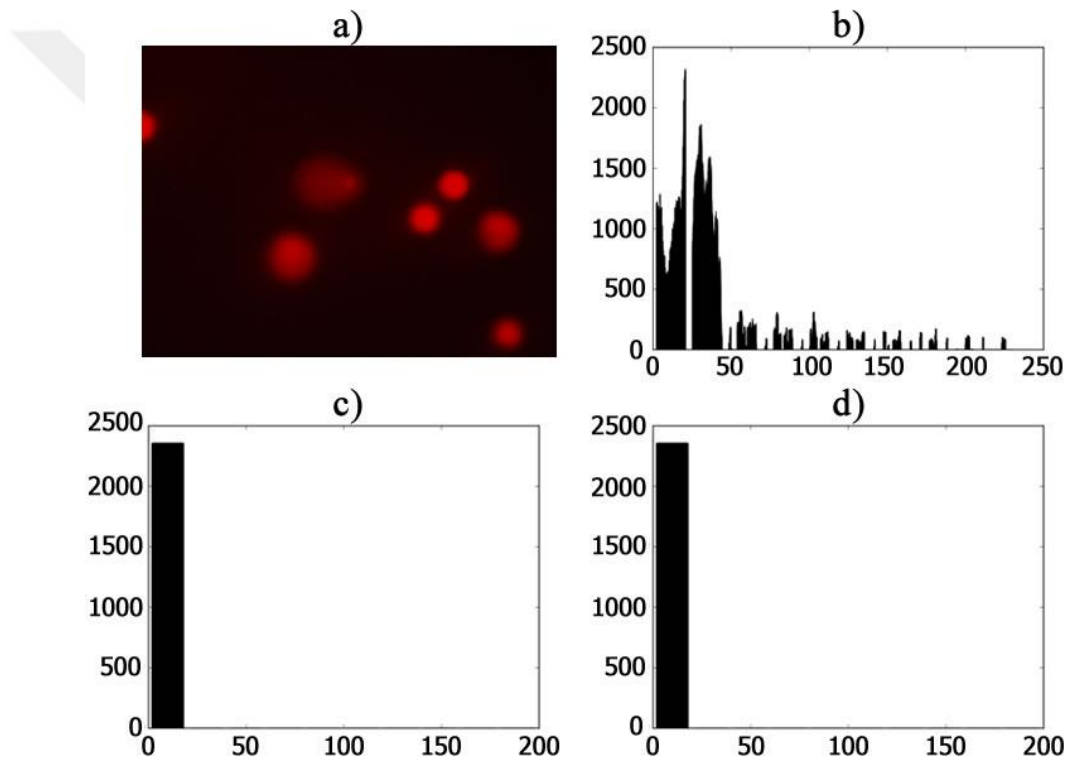


Figure 7.2. Comet assay image and histograms of channels. a) A sample comet assay image, b) Histogram of red channel image, c) Histogram of green channel image, d) Histogram of blue channel image.

### 7.1.1. Gaussian Filter

The obtained image is smoothed by using gaussian filter with a 5x5 kernel and median filter with a 5x5 kernel respectively. Gaussian filter, a low pass filter, is used to smooth images and remove noises and artefacts. Nevertheless, gaussian filter is more effective

at smoothing images. Its basis is like human visual perception system such that neurons form a similar filter in human brain when processing visual images. Gaussian function is given in Eq. 7.1 in one dimension and Eq. 7.2 in two dimensions [50,51].

$$G(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}} \quad (7.1)$$

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (7.2)$$

where  $\sigma$  is the standard deviation of the distribution. It is obtained by the standard deviation how much data are close to mean value. The formula of the standard deviation is given in Eq. 7.3 [50,51].

$$\sigma = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2} \quad (7.3)$$

where N is the number of elements,  $\bar{x}$  is arithmetic mean and  $x_i$  is i-th element.

Gaussian filter works with 2D Gaussian distribution function on images. Gaussian kernel coefficients are sampled from 2D Gaussian function. A kernel is walked throughout images. Intensity values of pixels in kernel are operated and a value is obtained. Intensity value of center pixel is replaced with the obtained value after this operation. A 5x5 convolution kernel approximating a Gaussian filter with  $\sigma = 1$  is given in Eq. 7.4 [50,51].

$$\frac{1}{273} \begin{array}{|c|c|c|c|c|} \hline 1 & 4 & 7 & 4 & 1 \\ \hline 4 & 16 & 26 & 16 & 4 \\ \hline 7 & 26 & 41 & 26 & 7 \\ \hline 4 & 16 & 26 & 16 & 4 \\ \hline 1 & 4 & 7 & 4 & 1 \\ \hline \end{array} \quad (7.4)$$

### 7.1.2. Median Filter

Median filter is a filter used to remove noises on images. The aim of median filter is to prevent oversize jumps among pixels in a kernel by taking median of intensity values of them. Then, intensity value of center pixel is replaced with median value of pixels in kernel. This process is performed on all pixels of image [52].

### 7.1.3. Thresholding Method

Thresholding has an important role at image segmentation applications due to perform intuitive features, simple realization and speed. A novel thresholding method has been developed in this thesis study. It is applied to convert obtained red channel images to binary images. To apply a thresholding method, a threshold value is needed. When a threshold value is obtained, the process is performed as given in Eq. 7.5 [50].

$$I(x, y) = \begin{cases} 0, & I(x, y) < T \\ 1, & I(x, y) \geq T \end{cases} \quad (7.5)$$

where  $T$  is a threshold value of image  $I$ . Calculation of an optimum threshold value is explained as follows. An iteration number is specified to read areas randomly on obtained image. Iteration number is calculated with the formula given in Eq. 7.6.

$$IterationNumber = \frac{ImageWidth * ImageHeight * Scale}{AreaWidth * AreaHeight} \quad (7.6)$$

AreaWidth and AreaHeight values are specified as 5. Scale factor is specified as 0.01. Because it is not obvious whether randomly read areas include head and tail parts of comets or not, a high iteration number is needed. An optimum scale factor is needed to prevent the application from decreasing speed and performance. Therefore, instead of walking on all areas of images, it is aimed to walk on a sufficient number of areas.

At each loop, x and y coordinates are randomly generated as the most top-left point of 5x5 area. Mean values of each row of areas are calculated in a cascade and binomial

way and then stored in a list. Thus, a 1x5 column vector is obtained. Mean value of the column vector is calculated in the same way. Thus, one possible thresholding value is obtained and stored in a list. This process is performed until *IterationNumber* is reached and possible thresholding values are obtained as a vector whose size is  $1 \times \text{IterationNumber}$ .

Value obtained by adding three terms is threshold value. First term is standard deviation of possible thresholding values stored in a list. Second term is arithmetic mean of possible thresholding values stored in a list. Third term is mean of possible thresholding values stored in a list calculated in cascade and binomial way. Let M be a randomly read 5x5 area and be shown as below.

$$M = \begin{bmatrix} 30 & 30 & 50 & 50 & 70 \\ 30 & 40 & 50 & 60 & 70 \\ 80 & 80 & 80 & 80 & 80 \\ 70 & 70 & 50 & 50 & 60 \\ 90 & 90 & 80 & 80 & 60 \end{bmatrix}$$

Mean values of each couple term in a row are calculated and stored. Then, obtained mean values are processed in same way. Thus, one term is diminished at each calculation loop. At the end, one value is obtained for a row. Same process is also performed for other rows. Number of obtained values are as many as row count. Same process is performed for all values on all rows inside other 5x5 areas as well. As a result, calculation process is completed. As an example, each step of calculation process of mean value for the first row in M is shown in Figure 7.3.

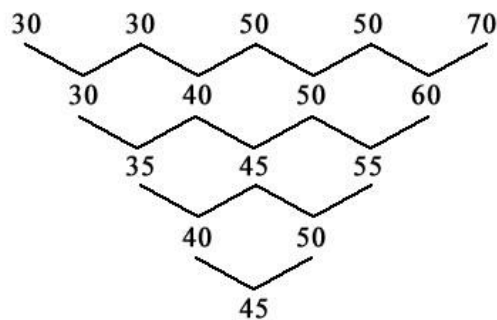


Figure 7.3. Calculation process of mean value of first row in M in cascade and binomial way.

45 is the cascade binomial mean value of first row in M. 50, 80, 56.875 and 81.875 are calculated respectively for other rows in M. When same process is performed for obtained five values, 64.649 as a threshold value of the randomly read area is obtained in a cascade and binomial way.

This process is performed on each randomly read 5x5 area until *iterationNumber* is reached and one value is obtained on each area and stored in a vector. Three different processes such as the process explained above, arithmetic mean and standard deviation are applied on the vector. The threshold value is obtained by adding results of the three processes. Images are converted to binary images by obtained threshold values in this method.

## **7.2. SEGMENTATION STAGE**

In this stage, the main aim is to extract all objects including healthy comets, non-healthy comets, blurry comets and overlapped comets on binary images and obtain all of them individually.

### **7.2.1. Connected Component Labeling**

Connected component labeling method is also known as connected component analysis, blob extraction, region labeling and region extraction in digital image processing area. By definition, pixel intensity values in black regions that stand for background are zero, pixel intensity values in white regions are one on binary images. Connected component algorithm is interested in detecting and labeling each different white region. Labeling is to assign different numbers to each individual region [53].

Connected component labeling consists of two stages as the first scan and the second scan from bottom to top and left to right. In the first scan, an equivalence relation is constructed among observed components. In the second scan, labels or different numbers are assigned to observed connected components. This equivalence relation in the first scan stores the topology of components used to label them in the correct way in the second scan [54].

In the first scan, every pixel is linearly checked one by one beginning from top-left corner to bottom-right corner. While checking pixels, 8-neighbors connectivity also known as Moore neighborhood and indirect neighbors is used and shown in Figure 7.4. If a pixel is a background pixel (its value is zero), it is simply ignored and moved on to next pixel. If a pixel does not belong to background, 8-neighbors connectivity of pixel is checked whether a neighbor exists or not. If there is no neighbor, a new label is assigned for current pixel. If there are neighbors with same label on them, same label with its neighbor is assigned for the current pixel. If there are neighbors with a different label on them, lower-valued label is assigned for current pixel. But, higher-valued label is stored as a child of lower-valued label. The first scan is completed until a label is assigned for last pixel [54].

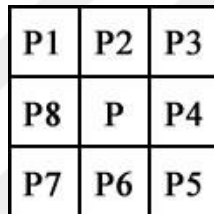


Figure 7.4. 8-neighbors connectivity.

The second scan is to rectify labels of connected components including different labels. Every pixel is linearly checked one by one beginning from top-left corner to bottom-right corner like the first scan. While checking label of a pixel, if it is not a child of any label, that label is assigned as root of itself and it is moved on to next pixel. If it is a child of any label, parent label is checked whether it is root of itself or not. If it is root of itself, child label is replaced with parent label. If it is not root of itself, its root is found, and child label is replaced with parent label as well. The second scan is completed until label of last pixel is rectified. As a result of this, connected component labeling is completed [54].

### 7.2.2. Removing Objects from Image Border

Now that each object has a label on binary image, it can be analyzed whether any object is located at border of image or not. Coordinates of borders of an image are obvious since image size is obvious. First row and column are located at zero index.



Last row and column are located at height of image and width of image respectively. If there exists a labeled object at one of these locations, pixel intensity values of that object are assigned as 0. As a result, these objects join among background pixels which are not interested in analysis.

### **7.2.3. Extracting Individual Objects**

Each white region that has a label must be extracted from binary images. Due to this, binary image of each individual object is obtained. Pixel intensity values of extracted binary image are multiplied with intensity pixel values of red channel image at corresponded locations. Thus, gray scale image of each individual object is obtained from binary image and is saved in loaded image folder as a new image whose size is equal to extracted object size. Its extension is same with extension of input image. Image name is given by starting from one to number of extracted objects.

## **7.3. ELIMINATION STAGE**

In this stage, the main aim is to obtain true objects and prevent the developed application from analyzing false objects, decreasing performance and time loss.

### **7.3.1. Elimination of Small Objects**

In this thesis study, areas of all true healthy comets are calculated, and it is empirically found that objects whose areas are lower than  $12000 \text{ px}^2$  are determined as noises and artefacts.  $12000 \text{ px}^2$  is the identifier value for  $3136 \times 2353$  images. Approximately  $12000 \text{ px}^2$  is 80% of mean value of all true individual healthy comet objects. Thus, intensity values of pixels inside areas lower than  $12000 \text{ px}^2$  are assigned to 0. As a result, small objects join among background pixels which are not interested in analysis.

### **7.3.2. Elimination of Blurry Objects**

In the literature, there are 36 different operators to detect blurry images [55]. Variance of Laplacian method, entropy of histogram method and gradient energy method are

three of them. In this thesis study, they are utilized to specify a solution for elimination of blurry individual comet objects.

### 7.3.2.1. Variance of Laplacian Method

This method is performed under three main stages. Firstly, Laplacian method is applied on individual grayscale comet object image. Secondly, variance of obtained image is calculated. Lastly, each individual comet object is classified as blurry or non-blurry based on a threshold value obtained by decision tree.

Laplace of a function with two variables like an image is calculated by the formula given in Eq. 7.7 [50,56].

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad (7.7)$$

This equation can be written by using finite difference method. First term of the equation can be rewritten in terms of x and given in Eq. 7.8. Second term of the equation can be rewritten in terms of y and given in Eq. 7.9 [50,56].

$$\frac{\partial^2 f}{\partial x^2} = f(x + 1, y) + f(x - 1, y) - 2f(x, y) \quad (7.8)$$

$$\frac{\partial^2 f}{\partial y^2} = f(x, y + 1) + f(x, y - 1) - 2f(x, y) \quad (7.9)$$

When Eq. 7.8 and Eq. 7.9 are replaced with the terms in Eq. 7.7, Laplace of a function with two variables is written and given in Eq. 7.10 [50,56].

$$\nabla^2 f = f(x + 1, y) + f(x - 1, y) + f(x, y + 1) + f(x, y - 1) - 4f(x, y) \quad (7.10)$$

Laplace of an image is calculated by a kernel walking on whole image. Laplacian kernel and its adapted forms used on images are shown in Figure 7.5 [50,56].

0	1	0
1	-4	1
0	1	0

0	-1	0
-1	4	-1
0	-1	0

1	1	1
1	-8	1
1	1	1

-1	-1	-1
-1	8	-1
-1	-1	-1

Figure 7.5. Laplace kernel to perform Eq. 7.10 and adapted forms.

Variance is one of the distribution parameters. It is defined as the average of squared differences from mean. Variance gives information how far a data set is spread out from mean value. Variance is the form of standard deviation without taking the square root. The formula of variance is given in Eq. 7.11 [55,57].

$$\sigma^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2 \quad (7.11)$$

Because comet object images have different sizes, the obtained variance values are divided by number of pixels on images. Thus, each comet object image is normalized.

### 7.3.2.2. Entropy of Histogram Method

This method is performed under three main stages. Firstly, histogram of grayscale comet object image is obtained. Secondly, entropy of obtained histogram is calculated. Lastly, each individual comet object is classified as blurry or non-blurry based on a threshold value obtained by decision tree. Entropy is explained in Chapter 7.4.5.3 in detail.

### 7.3.2.3. Gradient Energy Method

This method is performed under two stages. Firstly, sum of squares of the first derivative in the x and y directions is calculated [55]. Lastly, each individual comet object is classified as blurry or non-blurry based on a threshold value obtained by decision tree.

The formula of sum of squares of the first derivative in the x and y directions is given in Eq. 7.12 [55].

$$\varphi(x, y) = \sum_{i,j \in I(x,y)}^n ((I_x(i, j))^2 + (I_y(i, j))^2) \quad (7.12)$$

### 7.3.3. Elimination of Overlapped Comets

A novel method is developed to detect and eliminate overlapped comets. The flow chart of elimination of overlapped comets is shown in Figure 7.6.

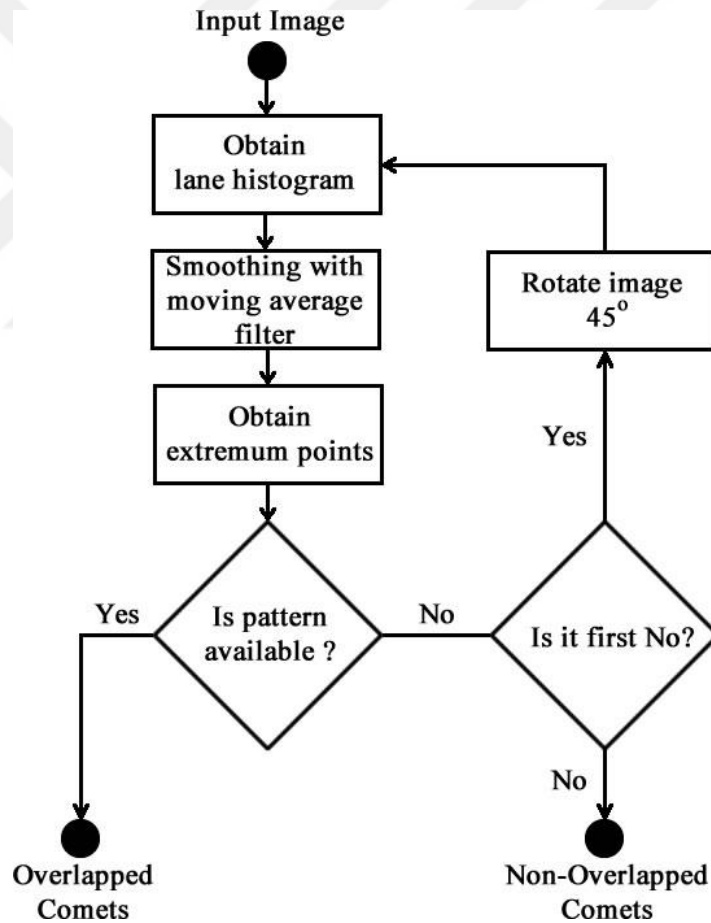


Figure 7.6. The flow chart of elimination of overlapped comets.

The developed application compares width of individual comet object image with height of individual comet object image. When width of image is greater than height of image, lane histogram throughout x axis of individual comet object image is

obtained. When height of image is greater than width of image, lane histogram throughout y axis of individual comet object image is obtained. The formula of calculation of lane histogram is given in Eq. 7.13 and Eq. 7.14.

$$lane[i] = \sum_{j=0}^{height-1} image[j, i] \quad (7.13)$$

$$lane[i] = \sum_{j=0}^{width-1} image[j, i] \quad (7.14)$$

After obtaining lane histogram of individual comet object image, the main aim is to find true maximum and minimum points and check whether one of specified patterns is detected respectively. Since overlapped comets have bright pixel values inside their own comet areas, lane histogram takes great values corresponding to areas including bright pixel values. As a result of this, hill shapes are located on lane histogram. On the other hand, overlapped comets have dark pixel values inside overlapped area. In addition to this, since overlapped area generally has a narrow area, that area has a small number of pixel values. This brings about that lane histogram take lower values than own comet areas and pit shape is located on lane histogram. Thus, specified pattern of overlapped comets includes at least two hills and one pit shape on lane histogram. Thus, it is proved that pattern is recognized by finding at least two maximum and one minimum points on lane histogram.

Lane histogram behaves like a time domain signal of individual comet object image including a lot of noises. That causes a lot of maximum and minimum points on lane histogram. To smooth signal and get rid of noises, moving average filter is applied with seven different window sizes as 24, 32, 48, 64, 96, 128 and 256 separately. The moving average filter is a very successful method to reduce random noises while retaining a sharp step response. That also makes the moving average filter successful for time domain signals. Since lane histogram behaves like time domain signals, the moving average filter is applied on lane histogram. The formula of the moving average filter is given in Eq. 7.15 [58].

$$y[i] = \frac{1}{M} \sum_{j=0}^{M-1} x[i + j] \quad (7.15)$$

M is moving window size. After signal is smoothed by moving average filter, maximum and minimum points of signal are ascertained by tracking increasing and decreasing values. While tracking increasing values, the higher value than following neighbor value is assigned as maximum point. While tracking decreasing values, the lower value than following neighbor value is assigned as minimum point. Tracking process continues until last value of signal. Pseudo code of this process is shown below.

```

for i from 0 to N-1
  if y[i] < y[i+1]
    min point list ← x[i],y[i]
  end
end

for i from 0 to N-1
  if y[i] > y[i+1]
    max point list ← x[i],y[i]
  end
end

```

N is number of points on signal. Standard deviation value of all ascertained maximum and minimum points is calculated since specified patterns is searched by using maximum points, minimum points and standard deviation value. First maximum point is directly appended to list. Then, if there exists a minimum point whose x coordinate is greater than addition of the x coordinate of appended maximum point and standard deviation value, the minimum point is appended to the list. Once again, if there exists a second maximum point whose x coordinate is greater than addition of the x coordinate of appended minimum point and standard deviation value, the maximum point is appended to the list. When the list consists of two maximum points which means two hills and one minimum point which means one pit, it stands for the fact that a specified pattern is generated and the observed comet is an overlapped comet. Pseudo code of this process is shown below.

```

list ←  $x_{\text{first max point}}$ 
for  $i$  in all min points
  if  $x_{\text{last appended max point}} + \text{std} < x_{\text{observed min point}}$ 
    list ←  $x_{\text{observed min point}}$ 
    break
for  $i$  in all max points
  if  $x_{\text{last appended min point}} + \text{std} < x_{\text{observed max point}}$ 
    list ←  $x_{\text{observed max point}}$ 
    break

if list size is 3
  return true
else
  return false

```

The pattern proved that there is an overlapped comet on individual comet object image must have one of the following types or similar to these types. Some pattern samples are shown in Figure 7.7.

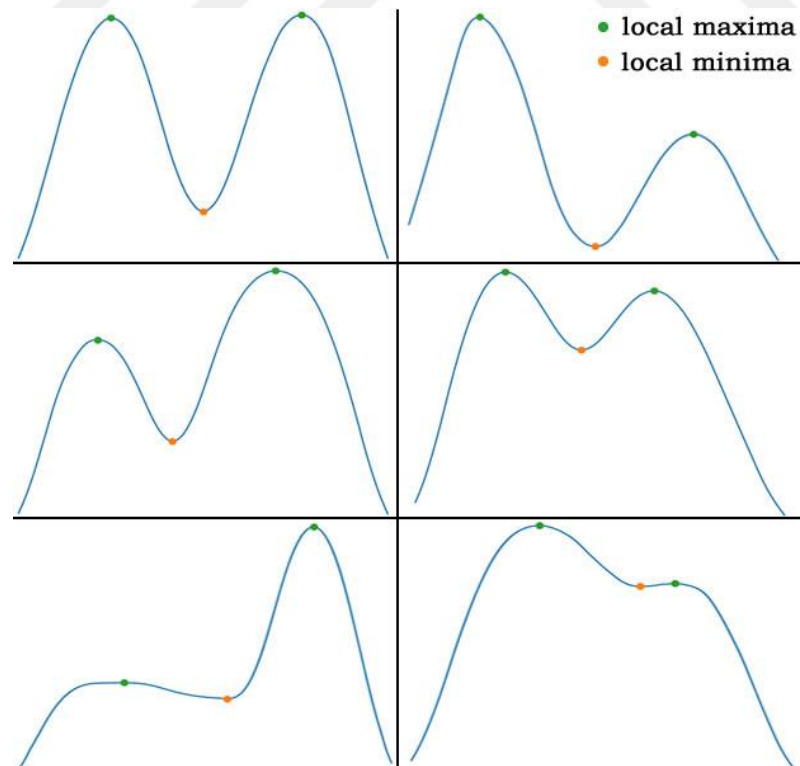


Figure 7.7. Pattern samples.

When a pattern is recognized on signal as one of the specified patterns or similar, it is classified as comets are overlapped comets. When a pattern is not recognized on signal as one of the specified patterns, it is checked whether signal is analyzed first time or not. If it is not first time analysis, it is classified as comets are non-overlapped comets. If it is first time analysis, individual comet image is rotated 45 degrees since lane histogram of crosswise overlapped comet is not consistent with one of the specified patterns. Same analysis is performed on 45 degrees rotated image.

#### **7.4. ANALYSIS STAGE**

In this stage, the main aim is to detect center of head part, detect tail direction, separate head part with tail part, calculate comet parameters as comet length, comet area, head length, head area, head percentage, tail length, tail area, tail percentage, tail moment and grade damage level.

##### **7.4.1. Detection of Center of Head Part**

Head part of DNA exists in case DNA with damage or DNA without damage. Tail part of DNA exists only when DNA has damage. While head part of DNA possesses bright pixel intensity values, tail part of DNA possesses darker pixel intensity values [2,41]. By using this information, the aim is to find sufficient brightest points on individual comet object image to detect center of head part.

Number of white pixels on binary individual comet object image stands for value of comet area. Coordinates of 1% brightest points in comet area are determined by ascending sorting intensity pixel values. Then, center of mass of them is calculated by taking arithmetic mean. Obtained coordinate is assigned as center of head part of individual comet object [2].

##### **7.4.2. Detection of Tail Direction**

If a comet has damage, the comet possesses a tail part. Detection of tail direction is needed to separate head part from tail part in a correct way. Image width and



coordinate of center of head part play an important role for detection of tail direction. If abscissas of center of head part (ACHP) is closer to (0, ordinate of center of head part (OCHP)) point, it means that tail direction keeps on the right-hand side. If x coordinate is closer to (width-1, OCHP) point, it means that tail direction keeps on the left-hand side.

### 7.4.3. Separating Head Part from Tail Part

After obtaining center of head part and tail direction, these two results are needed to separate head part from tail part. The closest contour point to center of head part which stands for radius of head part is calculated. With the distance formula between two points given in Eq. 7.16, head length is calculated.

$$distance = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (7.16)$$

Circle whose diameter is head length and center point is center of head part stands for head part. Other remaining white pixels belong to tail part. Thus, head part of individual comet object is separated from tail part of individual comet object by extracting circle.

### 7.4.4. Calculation of Comet Parameters

In this stage, calculation of comet parameters such as comet length, comet area, head length, head area, tail length, tail area, head percentage, tail percentage, and tail moment is explained.

Comet length is equal to value of extracted individual comet image width. Comet length is also equal to addition of head length and tail length [36,38].

Comet area is calculated by counting white pixels on extracted individual comet image. Comet area is also equal to addition of head area and tail area [36,38].

Detection of center of head part is explained in stage 7.4.1. The closest contour point to center of head part gives the radius of head part calculated with the distance formula between two points. Two times radius calculates diameter that stands for head length [36,38].

Because shape of head part is a circular shape, head area is calculated with formula of circle area given in Eq. 7.17 [36,38].

$$headArea = \pi \times \left(\frac{headLength}{2}\right)^2 \quad (7.17)$$

Head percentage is calculated by the formula given in Eq. 7.18 [36,38].

$$headPercentage = \frac{headArea}{cometArea} \times 100 \quad (7.18)$$

Tail length is calculated by the formula given in Eq. 7.19 [36,38].

$$tailLength = cometLength - headLength \quad (7.19)$$

Tail area is calculated by the formula given in Eq. 7.20 [36,38].

$$tailArea = cometArea - headArea \quad (7.20)$$

Tail percentage is calculated by the formula given in Eq. 7.21 [36,38].

$$tailPercentage = \frac{tailArea}{cometArea} \times 100 \quad (7.21)$$

Tail moment is calculated by the formula given in Eq. 7.22 [36,38].

$$tailMoment = tailLength \times tailPercentage \quad (7.22)$$

### **7.4.5. Grading Damage Level**

A novel method based on pixel profile analysis at vertical and horizontal directions based on center of head part, DTW and decision tree is developed to grade and classify individual comet objects such as G0, G1, G2 or G3.

#### **7.4.5.1. Pixel Profile Analysis**

The developed application performs pixel profile analysis at vertical and horizontal directions based on center of head part on individual comet object image. Horizontal pixel values at line segment passing over center of head part are stored in one list. Vertical pixel values at line segment passing over center of head part are stored in another list.

#### **7.4.5.2. Dynamic Time Warping (DTW)**

DTW is a matching method that finds an optimal alignment or similarity between two different time sequences under certain restrictions. The objective of DTW is to provide sequences with bringing same interval, match points to points and measure how similar sequences are. In the following, let first sequence be a pixel profile at horizontal direction based on center of head part of an individual comet object and denoted by  $X$  (length of  $X$ :  $x_1, x_2, \dots, x_n$  and  $n \in N$ ) and let second sequence be a pixel profile at vertical direction based on center of head part of same individual comet object and denoted by  $Y$  (length of  $Y$ :  $y_1, y_2, \dots, y_m$  and  $m \in N$ ). Lengths of  $X$  and  $Y$  do not have to be same for DTW [59,60]. An individual comet object sample has 452 horizontal and 200 vertical pixel intensity values and is shown in Figure 7.8. But, some pixel intensity values are presented in Table 7.1 because of large amount of data.

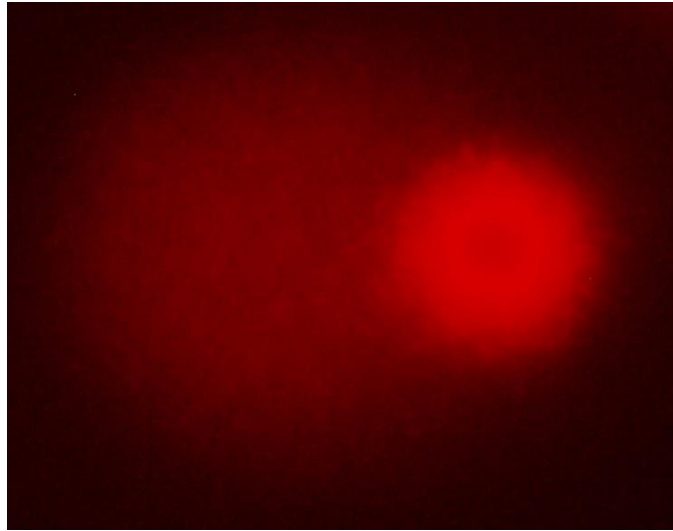


Figure 7.8. Individual comet object sample.

Table 7.1. Some pixel intensity values of an individual comet object sample.

Coordinate	Sequence X	Sequence Y
i	60	50
i+1	65	50
i+2	60	50
i+3	93	94
i+4	93	95
i+5	96	93
i+6	97	0
i+7	95	0
i+8	98	95
i+9	97	96
i+10	none	98

Any distance measurement algorithms like Euclidian, Manhattan etc. align i-th element of X with i-th element of Y. These algorithms give poorer similarity results than DTW. However, DTW aligns sequences much more successfully, finds warping

path and holds total distance as low as possible [59,60]. The results for the sequences X and Y presented in Table 7.1 are shown in Figure 7.9 when DTW is executed.

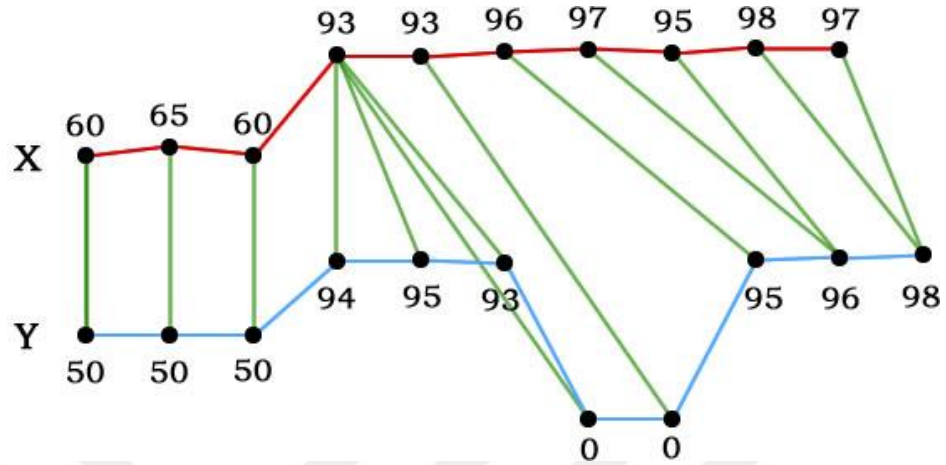


Figure 7.9. DTW results for the sequences X and Y.

While DTW takes two input parameters as X and Y, it returns cost matrix, minimum distance value and path as output. Output parameters can be decreased according to users' need. In cost matrix, distances between each element of X and each element of Y are stored. When there is an interval limit like  $\alpha$ , distances between  $i$ -th element of X and elements of Y interval at  $i-\alpha$  and  $i+\alpha$  are calculated and stored in cost matrix. Thus, performance of DTW increases with an optimal limit value. Minimum distance value is calculated by matching each  $i$ -th element of X with each nearest element of Y. The similarity of sequences is directly proportionate to low value of minimum distance [59,60]. Cost matrix for the sequences X and Y presented in Table 7.1 is presented in Table 7.2.

Table 7.2. Cost matrix for the sequences X and Y.

	60	65	60	93	93	96	97	95	98	97
50	10	25	35	78	121	167	214	259	307	354
50	20	25	35	78	121	167	214	259	307	354
50	30	35	35	78	121	167	214	259	307	354
94	64	59	69	36	37	39	42	43	47	50

<b>95</b>	99	89	94	38	38	38	40	40	43	45
<b>93</b>	132	117	122	38	38	41	42	42	45	47
<b>0</b>	192	182	177	131	131	134	138	137	140	142
<b>0</b>	252	247	237	224	224	227	231	232	235	237
<b>95</b>	287	277	272	226	226	225	227	227	230	232
<b>96</b>	323	308	308	229	229	225	226	227	229	230
<b>98</b>	361	341	346	234	234	227	226	229	227	228

The nearest element of Y for i-th element of X has to be minimum value of three values at cost matrix: one step right direction, one step top direction or one step top-right direction. It is shown in Figure 7.10.

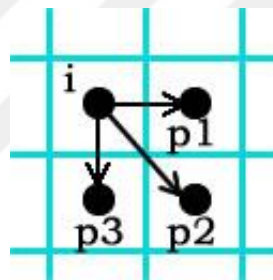


Figure 7.10. Possibilities of the nearest point for i-th element.

Sequences are nonlinearly warped by DTW. Path gives warped way between sequences. Path is formed by combining each single nearest point. Path is also called as warped function in DTW [59,60]. Path between the sequences X and Y according to pixel intensity values presented in Table 7.1 is shown in Figure 7.11.

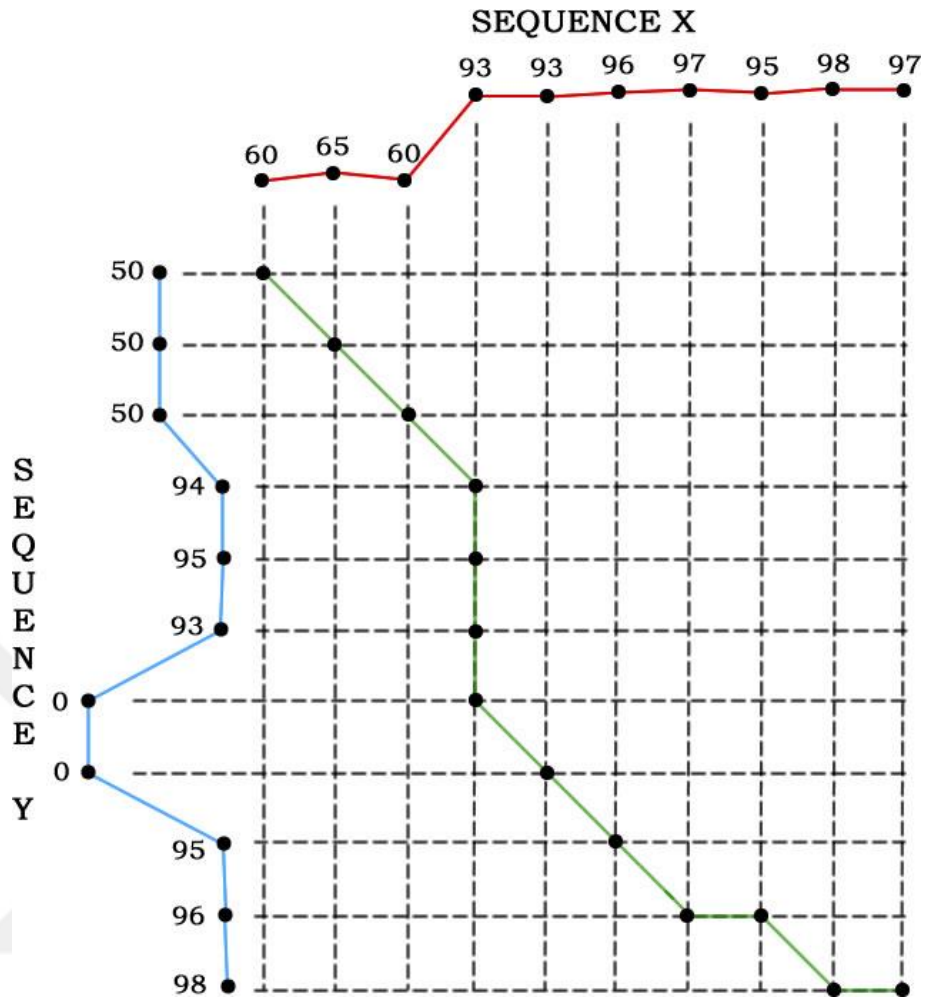


Figure 7.11. Path between the sequences X and Y.

At here, path is denoted by  $P$  that contains  $k$  points. It is denoted as  $P = P_1, P_2, \dots, P_s, \dots, P_k$ .  $s$ -th element of sequence X are matched with  $s$ -th element of sequence Y. Thus,  $P_s$  is written as  $P_s = (i_s, j_s)$ . DTW works to calculate minimum distance as follow:

```

distance = grid(1,1)
for i,j = (1,1) to (m,n)
    if grid(i,j-1) == grid(i-1,j) and grid(i-1,j) == grid(i-1,j-1)
        distance += grid(i,j-1)
    else
        distance += min(grid(i,j-1), grid(i-1,j), grid(i-1, j-1))
    end
end
end

```

There are five different restrictions on warping function such as monotone increment, continuity, boundary conditions, warping window and slope constraint in DTW [59,60].

Warping function should be an increasing function and must not go back in time index. This restriction ensures that it is not repeated in an alignment. Otherwise, alignment will not be an optimal and distance will be too high (monotone increment) [59,60].

Warping function should not include any jumping. It should be continuous at every point. This restriction ensures that alignment does not skip points (continuity) [59,60].

Warping function should start at bottom-left (1,1) and end at top-right (m,n). Otherwise, one sequence is compared with a part of another sequence (boundary condition) [59,60].

Warping function should not wander too far from diagonal of cost matrix. The best alignment occurs at path which is the closest to diagonal. This restriction ensures that alignment does not go to points including different features and takes points in specified window (warping window) [59,60].

Warping function should not include very long vertical and horizontal lines. This prevents path from moving away diagonal. This restriction prevents that short parts of sequences are matched to long parts of other sequences (slope constraint) [59,60].

### **7.4.5.3. Decision Tree**

Machine learning is a general definition of algorithms that overcome problems using samples obtained by problems. While some algorithms overcome problems based on prediction that can also be called as estimation, some overcome problems based on classification and clustering. Algorithms like k-means clustering, regression methods working with prediction learn from data and produce outputs. Algorithms like decision tree, support vector machines, k nearest neighboring working with classification determine a class for each sample [61-64].



Machine learning algorithms are divided into two classes such as supervised learning and unsupervised learning according to data structure. There exists a class concept in supervised learning. The main aim is to determine specific rules that separate classes from each other. There does not exist a class concept in unsupervised learning. The main aim is to discover and cluster similar samples in existing learning dataset [61-64].

Decision tree is one of the supervised machine learning algorithms that mainly classifies similar samples into same classes. Classification methods are divided into two classes such as algorithms based on entropy and classification and regression trees (CART). Iterative Dichotomiser 3 (ID3) decision tree and C4.5 decision tree belong to algorithms based on entropy. Twoing algorithm and Gini algorithm belong to CART [61-64]. C4.5 decision tree is an improved version of ID3 decision tree. The improvements are listed as follow:

- ✓ Numeric values are converted to binary values by a determined threshold value.
- ✓ Pruning process is performed.
- ✓ Branches can be carried to other levels of tree according to access frequency [61-64].

Decision trees are generally created by two steps such as tree creation and pruning. Creation tree is an iterative step. Decision tree starts with a root node and continues until leaf nodes. Disjunctive attributes are determined, become parent nodes and provide decision tree with branching. Decisions behave as leaf nodes. The iterative process is summarized as follow:

- ✓ Learning dataset is created.
- ✓ The best disjunctive attribute for samples in learning dataset is determined.
- ✓ One node is created by the best disjunctive attribute. Then, child nodes of this node are created. Samples in dataset belonging to child node are determined.
- ✓ For each sample determined in third step,
  - ✓ if all samples belong to same node, and
  - ✓ if there is no disjunctive attribute in dataset, and

- ✓ if there is no sample that carry values of rest attributes, process is terminated. If, not, continue with second step [61-64].

Information gain is calculated to determine each best disjunctive attribute. Entropy is used to calculate information gain. Entropy explains randomness, uncertainty and probability of an unhandled exception. Entropy is calculated as given in Eq. 7.23 [61-64].

$$E(X) = - \sum_{i=1}^n p(x_i) \log_2(p(x_i)) \quad (7.23)$$

$p(x_i)$  is the probability of  $x$  value in dataset  $X$  [63,65]. Information gain depending on entropy is calculated as given in Eq. 7.24 [61-64].

$$IG(X, attribute) = E(X) - \sum_{i=1}^n p(x_i) E(x_i) \quad (7.24)$$

Pruning is the second step that supplies removing noise data in dataset. If a node has a narrow range, pruning removes branches that brings about these nodes. As a result of this, complexity of trees and overfitting decreases. Pruning can start from root node or leaf node. Pruning consists of two methods such as pre-pruning and post-pruning. While pre-pruning is a method to prevent tree from growing early, post-pruning is applied after tree is completed [61-64].

Advantage of decision tree are listed as follow:

- ✓ Illustration, understanding and interpreting are easy.
- ✓ Both numeric and non-numeric values can be processed.
- ✓ Multioutput can be obtained.
- ✓ It is not influenced by non-linear and linear relations among attributes [61-64].

Disadvantages of decision tree are listed as follow:

- ✓ Complex trees can be created that cannot be generalized data.
- ✓ It is not guaranteed to give a best fit tree as an output.
- ✓ It is suggested to equalize class attributes in dataset before decision tree is performed [61-64].

In this thesis study, C4.5 decision tree based on entropy is utilized and performed by using MATLAB. 85% of all true comet objects are used as training set and 15% of all true comet objects are used as test set.

#### 7.4.5.4. Measurement Parameters of Decision Tree

Six different measurement parameters are obtained from each individual comet object to give decision tree. These parameters are root mean square error (RMSE), tail moment, ratio of ACHP with difference between width and ACHP, ratio of head length with tail length, ratio of head area with tail area and ratio of head percentage with tail percentage.

First parameter is RMSE calculated by using diagonal path and path obtained by DTW. RMSE is calculated by the formula given in Eq. 7.25 [66].

$$RMSE = \sqrt{\frac{\sum_{i=0}^{n-1} (x_{1,i} - x_{2,i})^2 + (y_{1,i} - y_{2,i})^2}{n}} \quad (7.25)$$

Diagonal path occurs when both sequences are equal. It is created by moving from origin to last abscissas of first sequence. Diagonal path must also validate  $y = x$  line. Although length of path and length of diagonal path are not same, number of their ordinates is same. Thus, horizontal distance is to calculate RMSE value between diagonal path and path obtained by DTW. Diagonal path and path obtained by DTW for an individual comet object are shown in Figure 7.12.

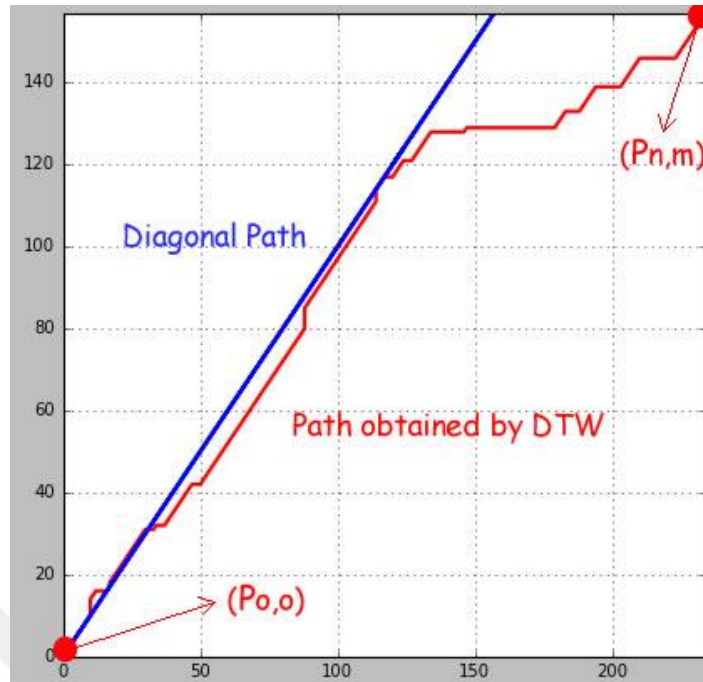


Figure 7.12. Diagonal path and path obtained by DTW.

When horizontal distance is taken into consideration, ordinates of both sequences are equal to each other. In other words,  $y_{1,i}$  is always equal to  $y_{2,i}$ . When diagonal path is taken into consideration, abscissas of diagonal path is always equal to ordinate of diagonal path. In other words,  $x_{2,i}$  is always equal to  $y_{2,i}$ . Thus, the formula of RMSE is modified as the formula given in Eq. 7.26.

$$Parameter1 = \sqrt{\frac{\sum_{i=0}^{n-1} (x_i - y_i)^2}{n}} \quad (7.26)$$

Second parameter is tail moment. The formula of tail moment is given in Eq. 7.27.

$$Parameter2 = tailLength \times tailPercentage \quad (7.27)$$

Third parameter is ratio of ACHP with difference between width and ACHP. Greater one locates at numerator part. Lower one locates at denominator part. The formula is given in Eq. 7.28.

$$Parameter3 = \frac{MAX(ACHP, width - ACHP)}{MIN(ACHP, width - ACHP)} \quad (7.28)$$

Fourth parameter is ratio of head length with tail length. The formula is given in Eq. 7.29.

$$Parameter4 = \frac{headLength}{tailLength} \quad (7.29)$$

Fifth parameter is ratio of head area with tail area. The formula is given in Eq. 7.30.

$$Parameter5 = \frac{headArea}{tailArea} \quad (7.30)$$

Sixth parameter is ratio of head percentage with tail percentage. The formula is given in Eq. 7.31.

$$Parameter6 = \frac{headPercentage}{tailPercentage} \quad (7.31)$$

## 7.5. PRESENTING RESULTS STAGE

A new folder named as loaded comet assay image name without its extension is created when an image is loaded to the developed application. In this folder, a loaded comet assay image, an excel file, each extracted individual comet object image and a pdf file for each individual comet object occupy. A loaded image shows id numbers, comet object contours and head contours drawn with green color. Id numbers of individual comet objects are written on center of head part. Excel file named as loaded image name includes values of comet parameters corresponding to each individual comet object row by row. Extracted individual comet object images whose names are their id numbers and extensions are same with loaded comet assay image extension show id number, comet object contours and head contours. Each pdf file is created for each individual comet object. A pdf file includes individual comet object image, id number, image width and height, x and y coordinates of center of head part, values of comet

parameters and file created time. Samples of the mentioned files are shown in Figure 7.13-7.17.

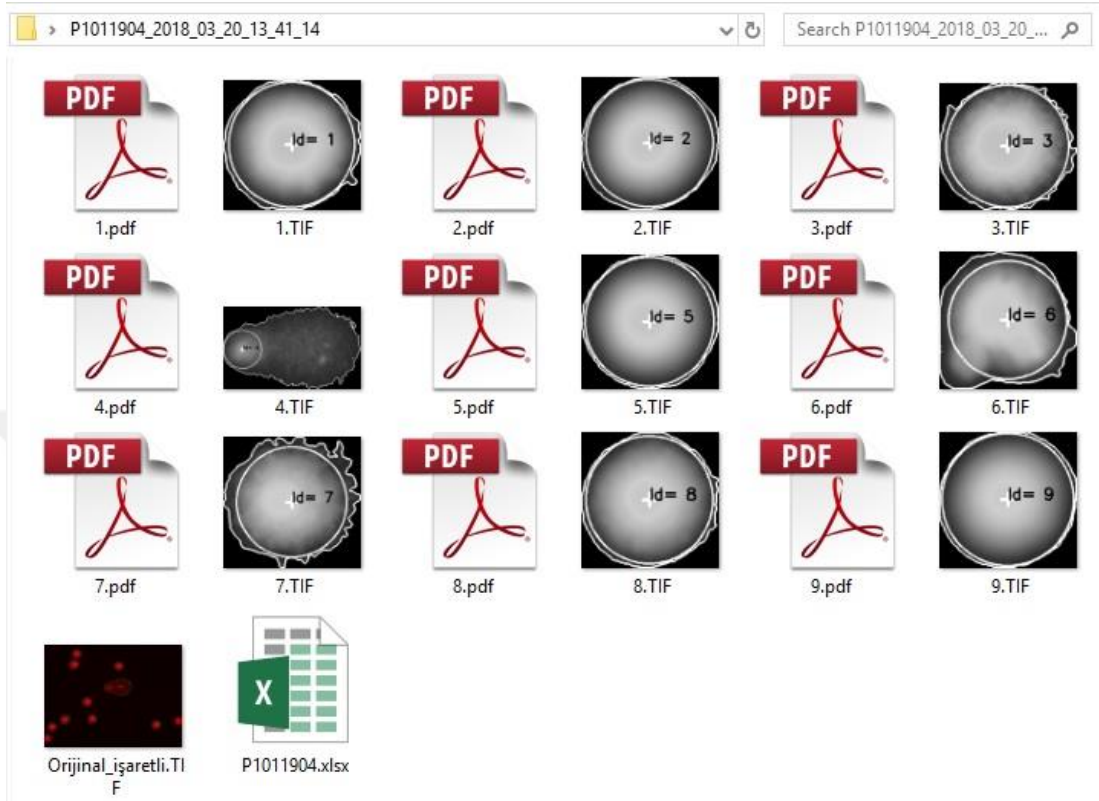


Figure 7.13. A sample view of a folder.

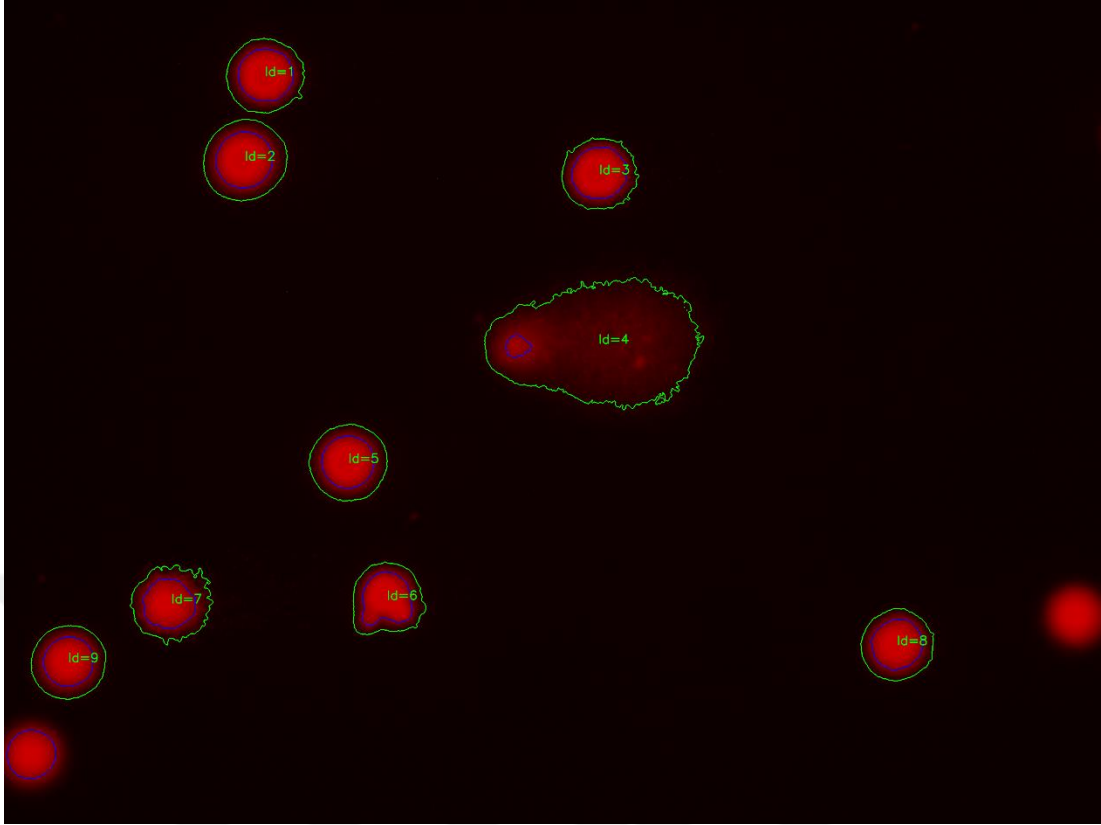


Figure 7.14. A sample loaded comet assay image.

P1011904.xlsx - Excel

efalsehirli@karabuk.edu.tr

Dosya Giriş Ekle Sayfa Düzeni Formüller Veri Gözetim Geçir Görünüm Eklenimler LOAD TEST Team Ne yapmak istediğinizi söyleyin

Yapıştır Kopyala Biçim Boyayıcı Pano Yazı Tipi Hizalama Sayı

Metni Kaydır Genel Koşullu Tablo Olarak Hücre Biçimlendime Biçimlendir Stilleri Ekle Sil Biçim Doldur Sırala ve Filtre Bul ve Seç Otomatik Toplam Doldur Temizle Düzenleme

KOMET ID	HEAD MERKEZ	HEAD UZUNLUK	TAIL UZUNLUK	HEAD ALAN	TAIL ALAN	HEAD YÜZDE	TAIL YÜZDE	TAIL MOMENT	KOMET UZUNLUK	KOMET ALAN	HASAR SINIFI
1	114.00 Y: 100.0	192.42	30.58	29078.58	7033.42	80.52	19.48	595.67	223.00	36112.00	Saglıkk
2	113.00 Y: 114.0	215.08	24.92	36332.51	6375.49	85.07	14.93	371.99	240.00	42708.00	Saglıkk
3	102.00 Y: 101.0	188.35	31.65	27862.78	5848.22	82.65	17.35	549.05	220.00	33711.00	Saglıkk
4	X: 88.00 Y: 194.00	172.29	452.71	23313.75	139818.25	14.29	85.71	38801.13	625.00	163132.00	Ağır Hasarlı
5	X: 109.00 Y: 109.0	209.48	14.52	34463.26	2883.74	92.28	7.72	112.15	224.00	37347.00	Saglıkk
6	X: 80.00 Y: 83.00	145.34	61.66	16590.75	16741.25	49.77	50.23	3096.87	207.00	33332.00	Saglıkk
7	X: 102.00 Y: 105.0	175.93	60.07	24309.64	12835.36	65.45	34.55	2075.64	236.00	37145.00	Saglıkk
8	X: 99.00 Y: 97.00	179.89	28.11	25415.48	7057.52	78.27	21.73	610.95	208.00	32473.00	Saglıkk
9	X: 99.00 Y: 100.00	188.73	23.27	27975.88	6782.12	80.49	19.51	454.00	212.00	34758.00	Saglıkk

P1011904

Figure 7.15. A sample excel file for a loaded comet assay image.

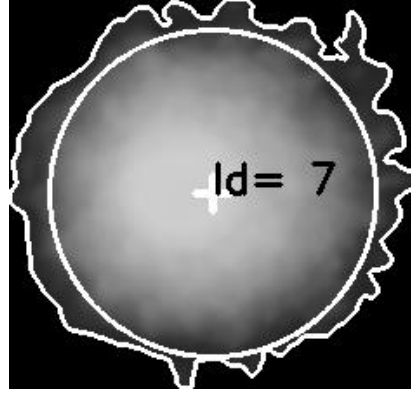
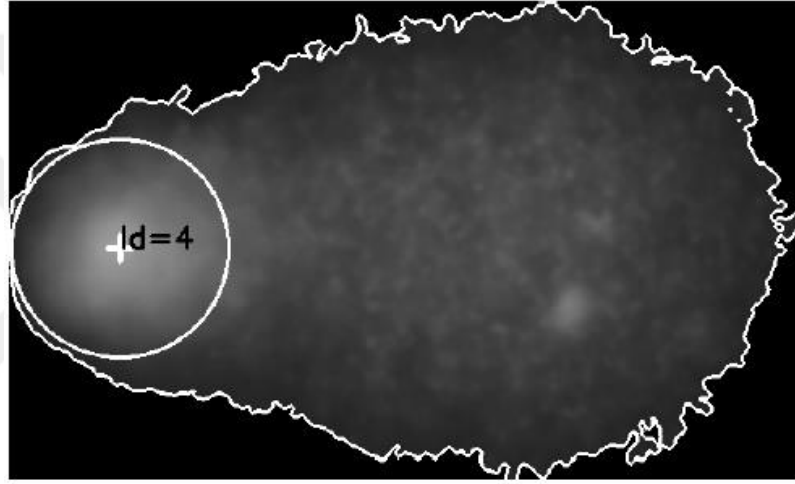


Figure 7.16. A sample individual comet object.



Komet Id: 4  
Resim Genişlik: 625  
Resim Yükseklik: 377  
Head Merkez Noktası: X: 88.00 Y: 194.00  
Head Uzunluğu: 172.29  
Tail Uzunluğu: 452.71  
Head Alanı: 23313.75  
Tail Alanı: 139818.25  
Head Yüzdesi: 14.29  
Tail Yüzdesi: 85.71  
Tail Momenti: 38801.13  
Komet Uzunluğu: 625.00  
Komet Alanı: 163132.00  
Hasar Sınıfı: Ağır Hasarlı

2018-07-04 16:25:30

Figure 7.17. A sample pdf file for an individual comet object.



## 7.6. VALIDATION

2476 comet assay images were used in this thesis study. All images were evaluated by a clinical expert. All comet objects were classified as blurry or non-blurry objects. All comet objects were classified as overlapped or non-overlapped comets. All comet objects were classified as G0, G1, G2 or G3. The developed application also classifies comet objects automatically. Results of the developed application were compared with clinical expert's results and validated based on confusion matrix.

Confusion matrix, also called as an error matrix, provides visualization of performance of developed applications, algorithms or methods in a table [62]. It mainly consists of four parameters such as True Positive (TP), False Positive (FP), False Negative (FN) and True Negative (TN). TP stands for correctly identified conditions. FP stands for incorrectly identified conditions. FN stands for incorrectly rejected conditions. TN stands for correctly rejected conditions [67,68]. In more detail, TP is the number of detected comet objects counted as comet objects, FP is the number of detected non-comet objects counted as comet objects, FN is the number of detected comet objects counted as non-comet objects and TN is the number of detected non-comet objects counted as non-comet objects.

Sensitivity refers to the ability of the developed application to correctly detect comet objects evaluated as comet objects by clinical expert. In other words, sensitivity is a probability of a positive test given that object is a comet object. Sensitivity is calculated by the formula given in Eq. 7.32 [69].

$$Sensitivity = \frac{TP}{TP + FN} \quad (7.32)$$

Specificity relates to the ability of the developed application to correctly reject non-comet objects evaluated as non-comet objects by clinical expert. In other words, specificity is probability of a negative test given that object is non-comet object. Specificity is calculated by the formula given in Eq. 7.33 [69].

$$Specificity = \frac{TN}{TN + FP} \quad (7.33)$$

Accuracy is the proximity of obtained results by the developed application to true value. In other words, accuracy is probability of a test given that comet object is counted as comet object and non-comet object is counted as non-comet object. Accuracy is calculated by the formula given in Eq. 7.34 [68].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (7.34)$$

Confusion matrix is presented in Table 7.3.

Table 7.3. Confusion matrix.

		Expert Side	
		Condition Positive	Condition Negative
Application Side	Application produces positive	TP	FP
	Application produces negative	FN	TN
		<i>Sensitivity</i> $\frac{TP}{TP + FN}$	<i>Specificity</i> $\frac{TN}{TN + FP}$

In statistics, receiver operating characteristic (ROC) graph or curve is a graphical illustration for the test based on different variables on same data. The ROC graph is generated by different observed variables used to determine which values of the observed variables will be considered abnormal. Then, sensitivity values of obtained results against corresponding false positive rates are plotted [70,71].

## CHAPTER 8

### RESULTS

In this thesis study, comet assay experiments have been performed, comet assay images have been obtained at the end of experiments, a fully automated desktop application has been developed with Python programming language and obtained comet assay images have been analyzed and quantified by the developed application.

9206 individual comet assay objects have been extracted from more than 2476 original images. All individual comet assay objects have been analyzed and quantified based on elimination conditions, calculation of comet parameters and grading damage level by the developed application. Non-comet objects are eliminated and not analyzed according to size, blurry and overlapped conditions. True comet objects are not eliminated and they are accepted. Comet parameters are calculated by the help of their formulas. Each individual comet object is graded such as G0, G1, G2 or G3 based on measurement parameters.

Comet IV is a commercial application used for analysis of comet assay images. When obtained comet assay images at the end of experiments have been tested on this application, it is determined that Comet IV does not eliminate blurry and overlapped comets and grade damage level. Hence, the developed application is not compared with Comet IV.

OpenComet is an open source application used for analysis of comet assay images. When obtained comet assay images at the end of experiments have been tested on this application, it is determined that OpenComet does not eliminate blurry comet objects and grade damage level. However, OpenComet eliminates overlapped comets. Hence, the developed application is compared with OpenComet about elimination of overlapped comets.

This chapter presents results obtained by the developed application.

## **8.1. ELIMINATION RESULTS**

Each non-comet object is eliminated according to size, blurry and overlapped conditions.

### **8.1.1. Elimination of Small Objects**

427 small individual comet objects have been evaluated. 202 objects locating at borders of images have been evaluated. All of them are eliminated, assigned their pixels as 0 and join among background pixels. 100% success rate is obtained.

### **8.1.2. Elimination of Blurry Objects**

Seven different parameters are separately performed to eliminate blurry object on individual comet object images such as variance of Laplacian, entropy of histogram, gradient energy, multiplication of variance of Laplacian with entropy of histogram, multiplication of entropy of histogram with gradient energy, multiplication of variance of Laplacian with gradient energy and multiplication of all. These seven parameters are separately utilized in decision tree to classify objects as blurry or non-blurry. Decision tree structures for each parameter are shown in Figure 8.1-8.7. The comparison of each parameter based on sensitivity, specificity and accuracy according to clinical expert results is presented in Table 8.1.

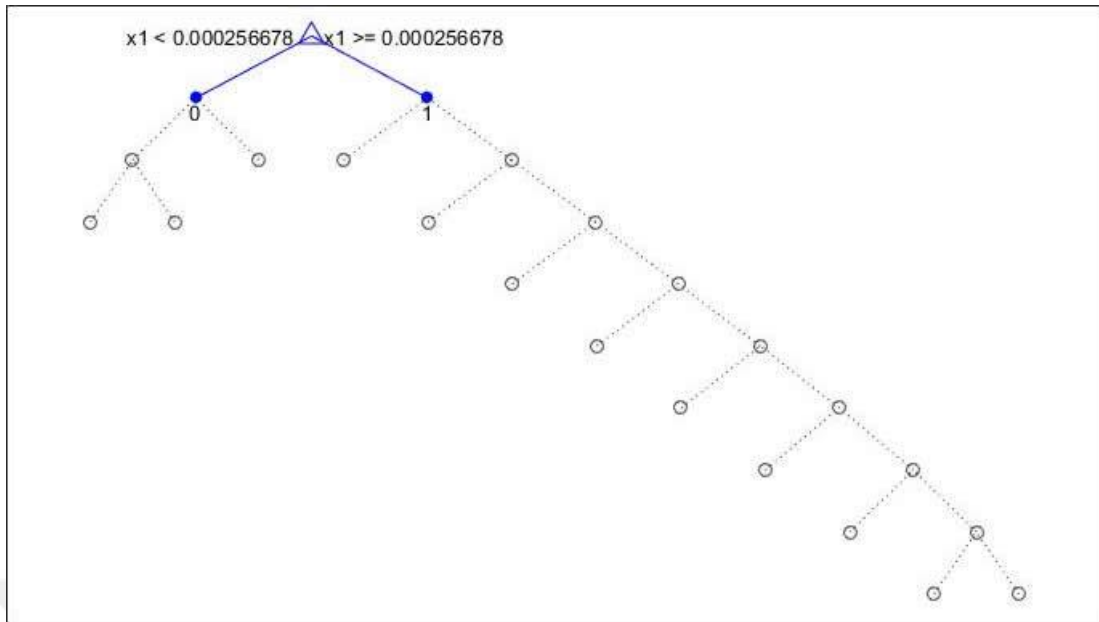


Figure 8.1. Decision tree structure obtained by variance of Laplacian with 2-level pruning.

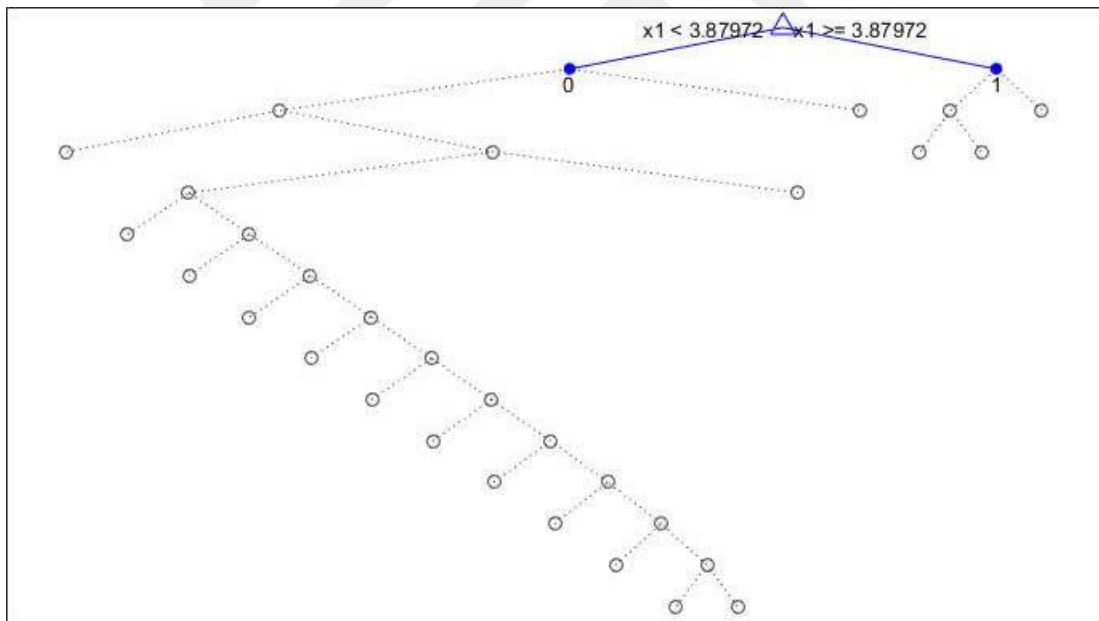


Figure 8.2. Decision tree structure obtained by entropy of histogram with 2-level pruning.

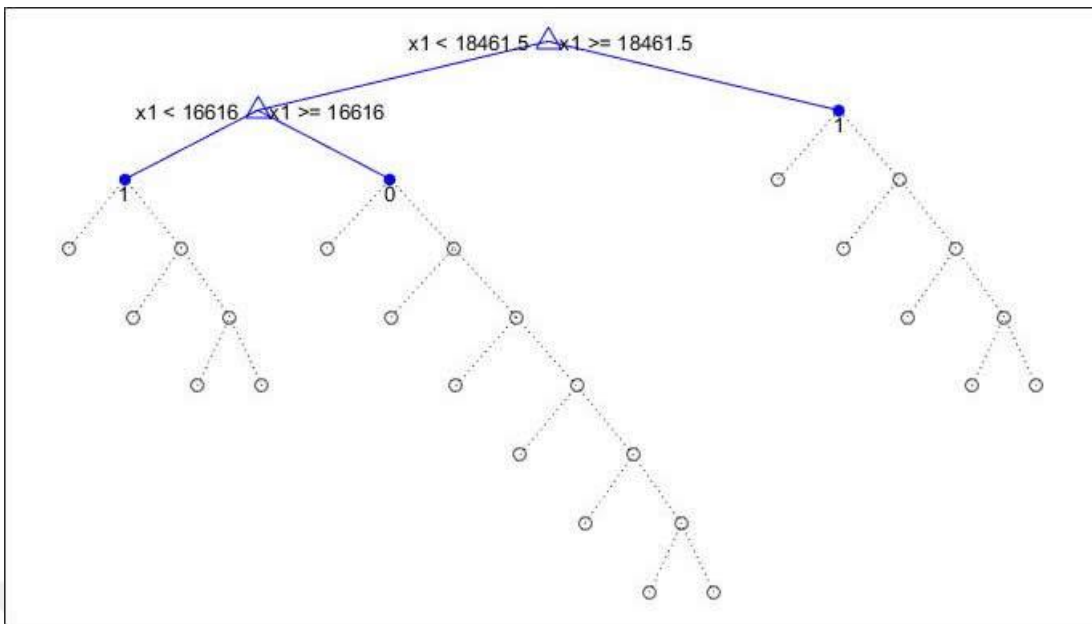


Figure 8.3. Decision tree structure obtained by gradient energy with 4-level pruning.

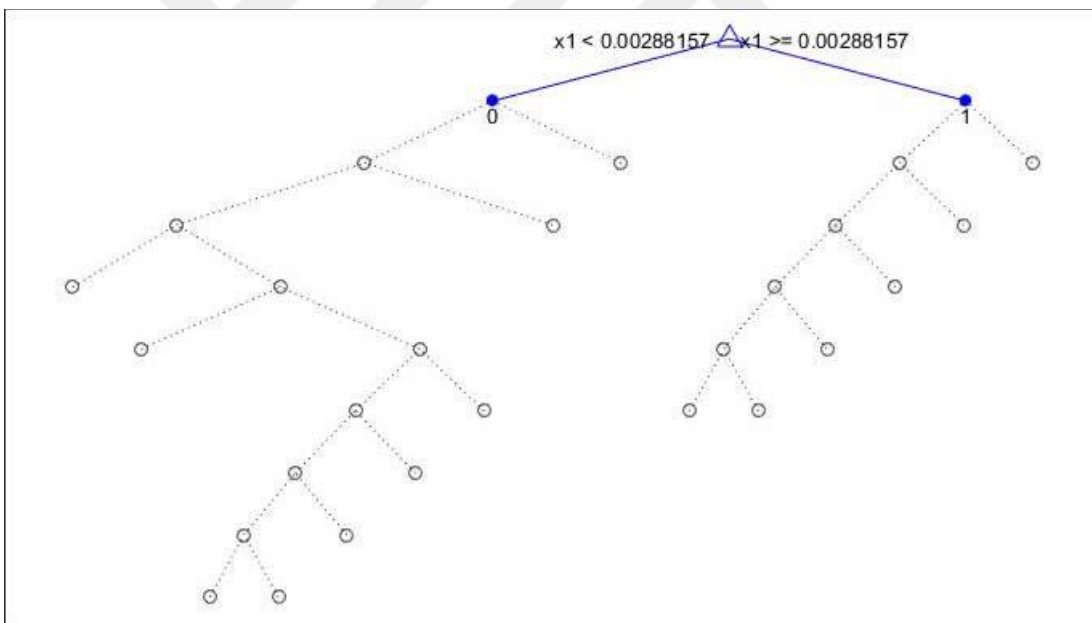


Figure 8.4. Decision tree structure obtained by multiplication of variance of Laplacian with entropy of histogram with 3-level pruning.

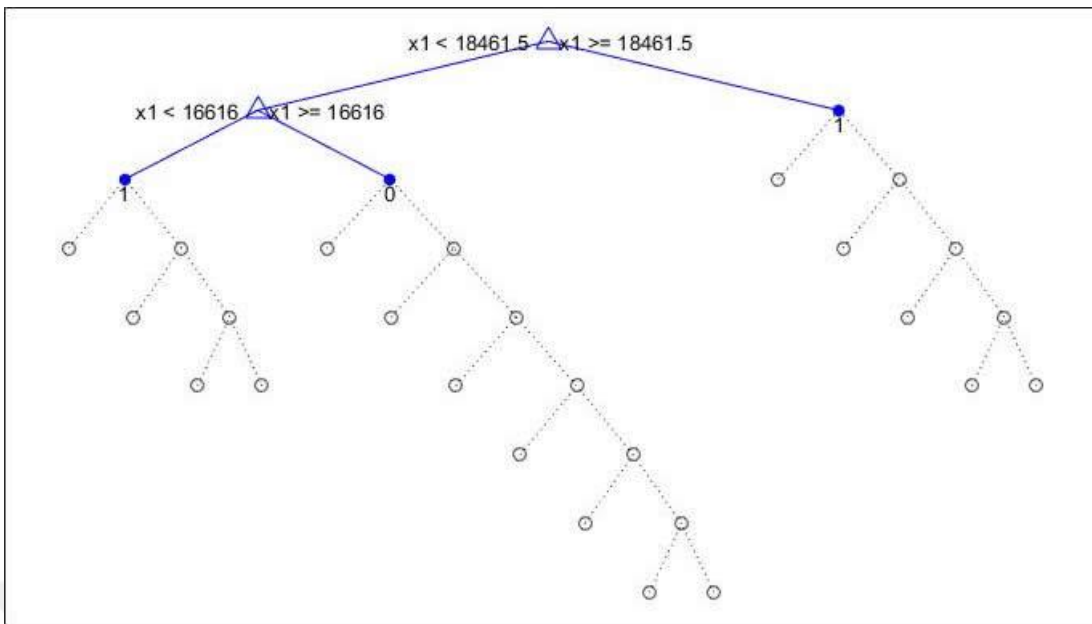


Figure 8.5. Decision tree structure obtained by multiplication of entropy of histogram with gradient energy with 3-level pruning.

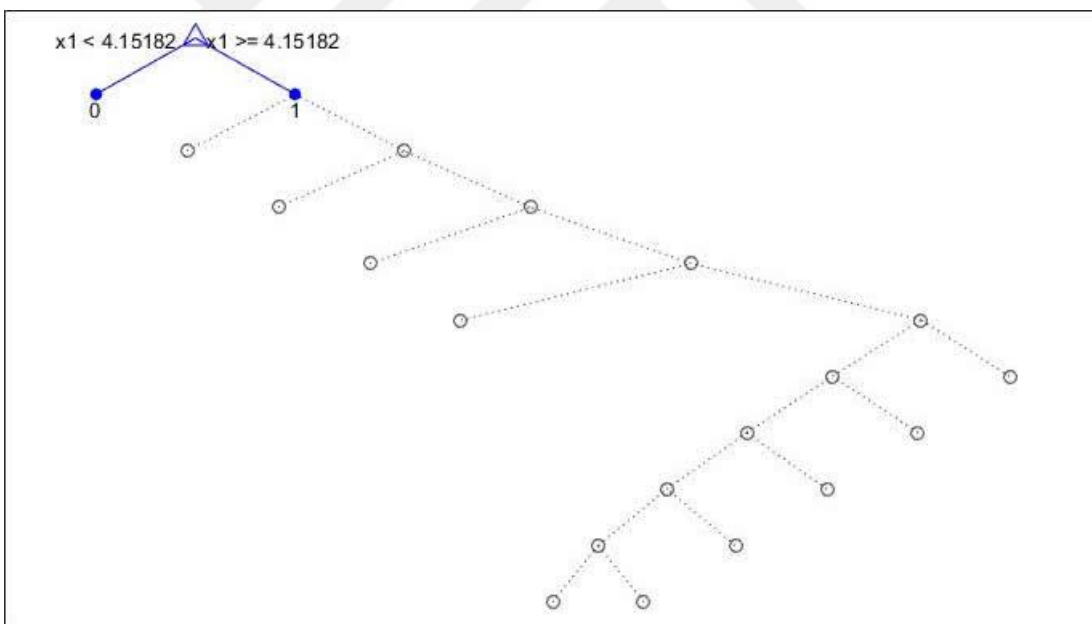


Figure 8.6. Decision tree structure obtained by multiplication of variance of Laplacian with gradient energy with 2-level pruning.

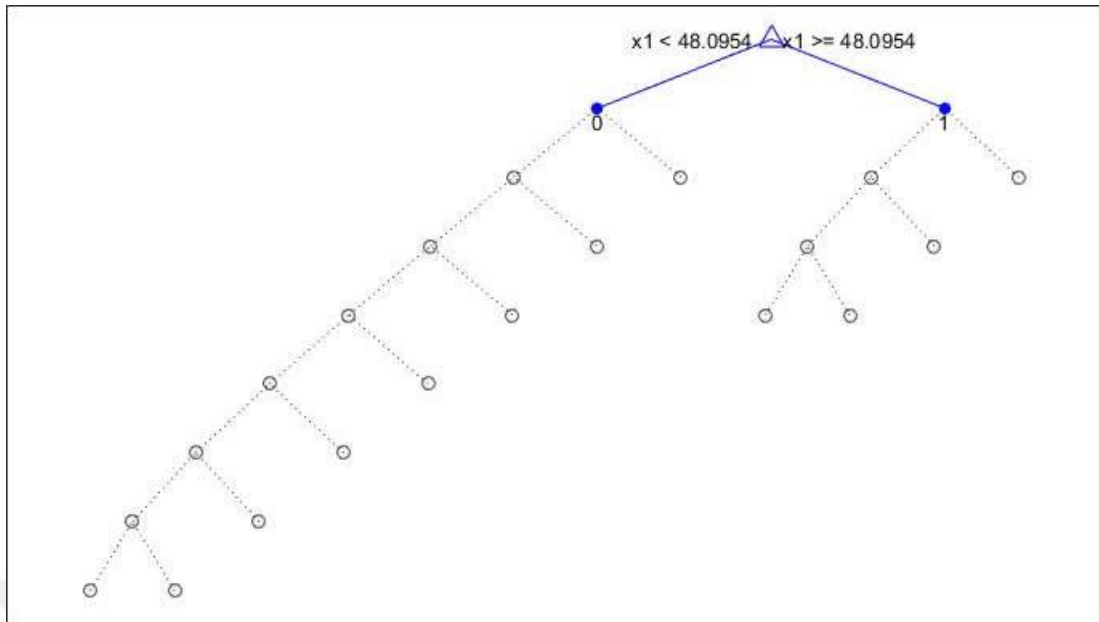


Figure 8.7. Decision tree structure obtained by multiplication of all with 4-level pruning.

Table 8.1. Validation of parameters for elimination of blurry objects.

Method	Sensitivity (%)	Specificity (%)	Accuracy (%)
variance of Laplacian	95.83	21.43	48.48
entropy of histogram	55.56	86.67	69.70
gradient energy	83.33	43.33	65.15
variance of Laplacian x entropy of histogram	61.11	63.33	62.12
entropy of histogram x gradient energy	91.67	50.00	72.73
variance of Laplacian x gradient energy	88.89	30.00	62.12
variance of Laplacian x entropy of histogram x gradient energy	61.11	63.33	62.12

Even though seven different parameters are separately utilized in decision tree to classify objects as blurry or non-blurry, it is needed to improve success rate based on sensitivity, specificity and accuracy. Hence, seven parameters are used all together in



decision tree to classify objects. Sensitivity, specificity and accuracy are calculated as 72.22%, 93.33% and 81.82 respectively. Decision tree structure for elimination of blurry comet objects is shown in Figure 8.8.

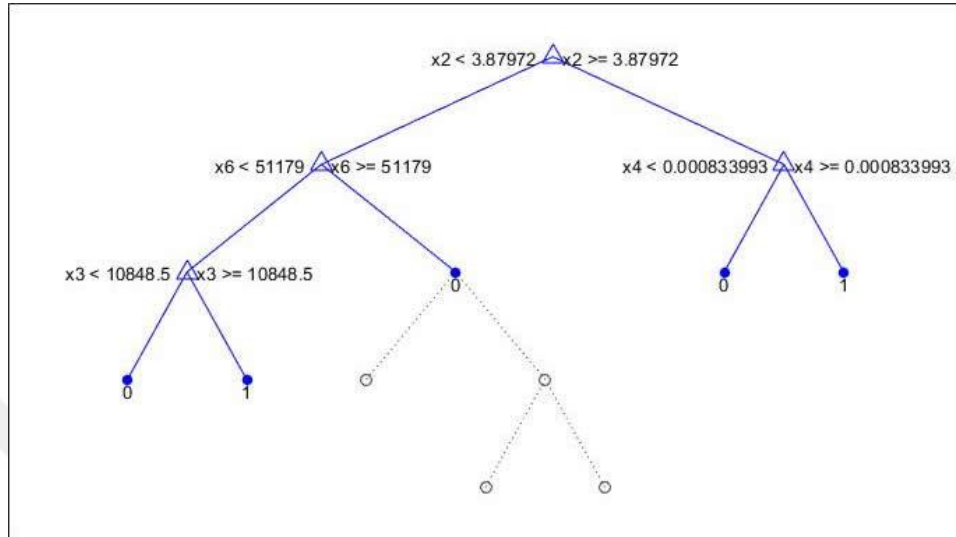


Figure 8.8. Decision tree structure with 1-level pruning for elimination of blurry comet objects.

### 8.1.3. Elimination of Overlapped Comets

Individual comet object images, obtained lane histograms, smoothed lane histograms and ROC graph of each window size of moving average filter are presented in this stage.

In this thesis study, seven different window sizes of moving average filter are used to compare and find the best one among them. The best result is obtained based on sensitivity, specificity, accuracy and ROC graph when window size is 96.

Five sample overlapped comets, their lane histograms and their smoothed lane histograms including local maximum and minimum points are shown in Figure 8.9-8.13 when window size is 96. Validation for each window size of moving average filter based on sensitivity, specificity, accuracy and covered area under curve on ROC graph according to clinical expert results is presented in Table 8.2. The obtained ROC graph for each window size of moving average filter is shown in Figure 8.14.

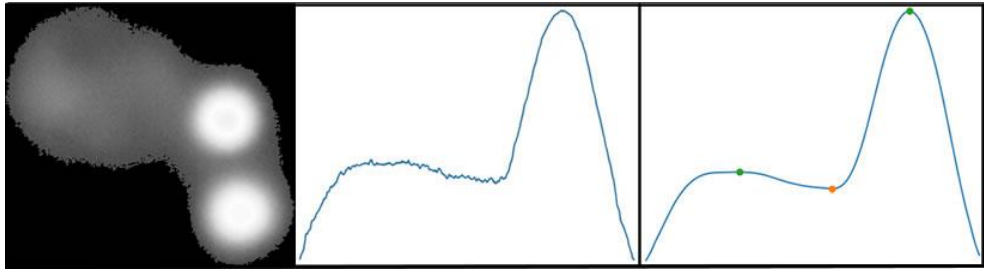


Figure 8.9. A sample overlapped comet, its lane histogram and its smoothed lane histogram.

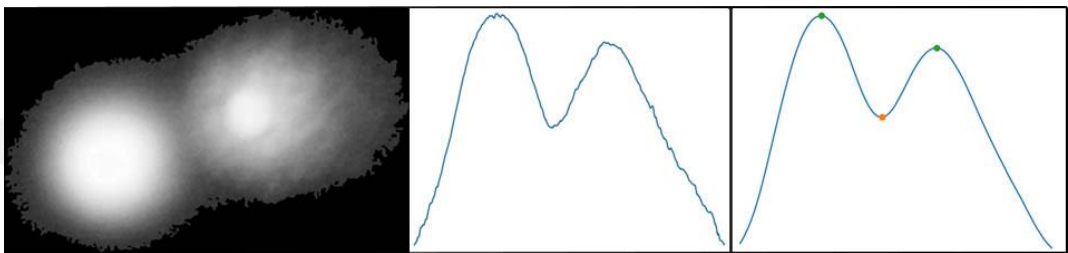


Figure 8.10. A sample overlapped comet, its lane histogram and its smoothed lane histogram.

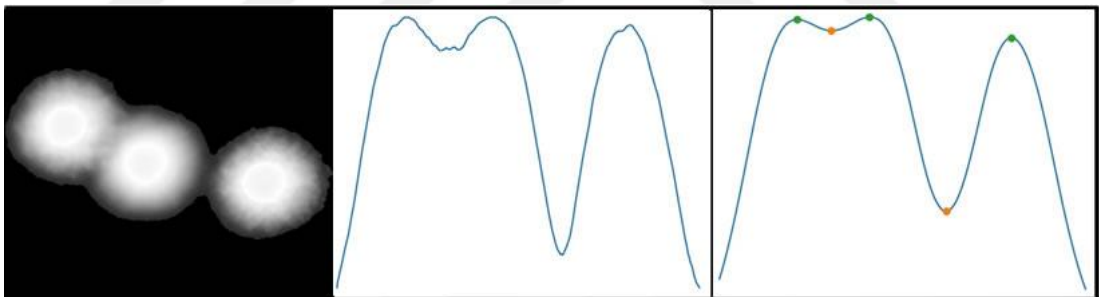


Figure 8.11. A sample overlapped comet, its lane histogram and its smoothed lane histogram.

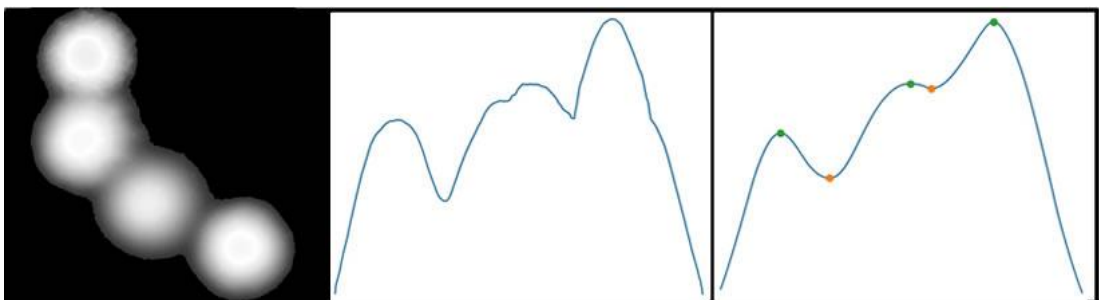


Figure 8.12. A sample overlapped comet, its lane histogram and its smoothed lane histogram.

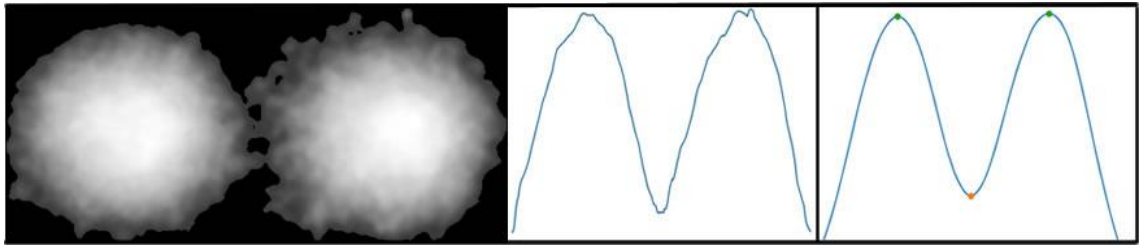


Figure 8.13. A sample overlapped comet, its lane histogram and its smoothed lane histogram.

Table 8.2. Validation for seven different window sizes of moving average filter.

Window size	Sensitivity (%)	Specificity (%)	Accuracy (%)	Area (%)
24	92.00	71.91	74.85	82
32	100	80.14	83.04	90
48	92.00	86.97	87.72	89
64	91.30	87.16	87.72	89
96	91.30	93.24	92.98	92
128	81.48	93.06	91.23	87
256	60.00	95.89	90.64	78

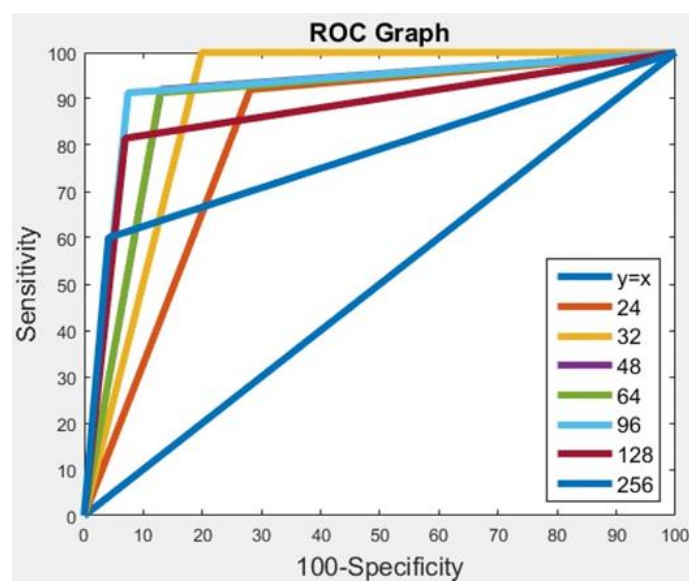


Figure 8.14. ROC graph of seven different window sizes of moving average filter.

Detection of overlapped and non-overlapped comets has been compared with only OpenComet which is an open source application using same comet assay images. OpenComet eliminates overlapped comets a bit more successfully than the developed application according to specificity and accuracy. However, there is a dramatical difference for the benefit of the developed application according to sensitivity. The comparison between OpenComet and the developed application based on sensitivity, specificity and accuracy is presented in Table 8.3.

Table 8.3. The comparison between OpenComet and the developed application.

<b>Application</b>	<b>Sensitivity (%)</b>	<b>Specificity (%)</b>	<b>Accuracy (%)</b>
OpenComet	62.96	100.00	94.12
The developed application	91.30	93.24	92.98

## 8.2. GRADING RESULTS

Individual comet object images, pixel profile analysis results, path obtained by DTW, measurement parameters for decision tree and results of the method are presented in this stage. Individual comet objects of each damage level, vertical pixel profiles, horizontal pixel profiles and path obtained by DTW are shown in Figure 8.15-8.18. Six measurement parameters for each comet object in Figure 8.15-8.18 are presented in Table 8.4. Interval of six measurement parameters for each damage level is presented in Table 8.5. Confusion matrices based on accuracy, sensitivity and specificity for each damage level are presented in Table 8.6-8.7.

Six measurement parameters are used for decision tree to grade DNA damage with high sensitivity, specificity and accuracy. Although parameter1, parameter2, parameter3 and parameter6 are calculated, parameter2 and parameter5 are enough and very powerful classifiers. Thus, decision tree does not need them to grade DNA damage. Decision tree structure to grade damage level is shown in Figure 8.19. When other comet assay images used in this thesis study are tested, expected results are obtained.

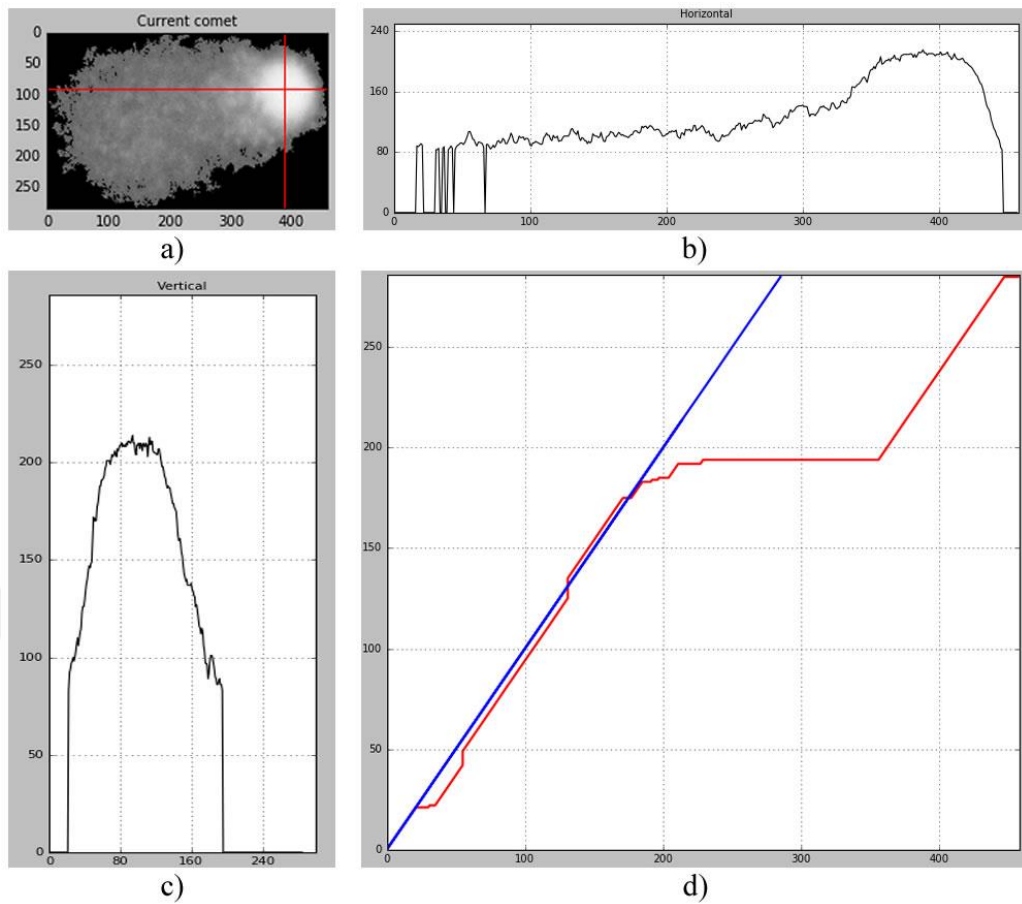


Figure 8.15. G3 comet object. a) Comet image. b) horizontal pixel profile analysis. c) vertical pixel profile analysis. d) diagonal path and path obtained by DTW.

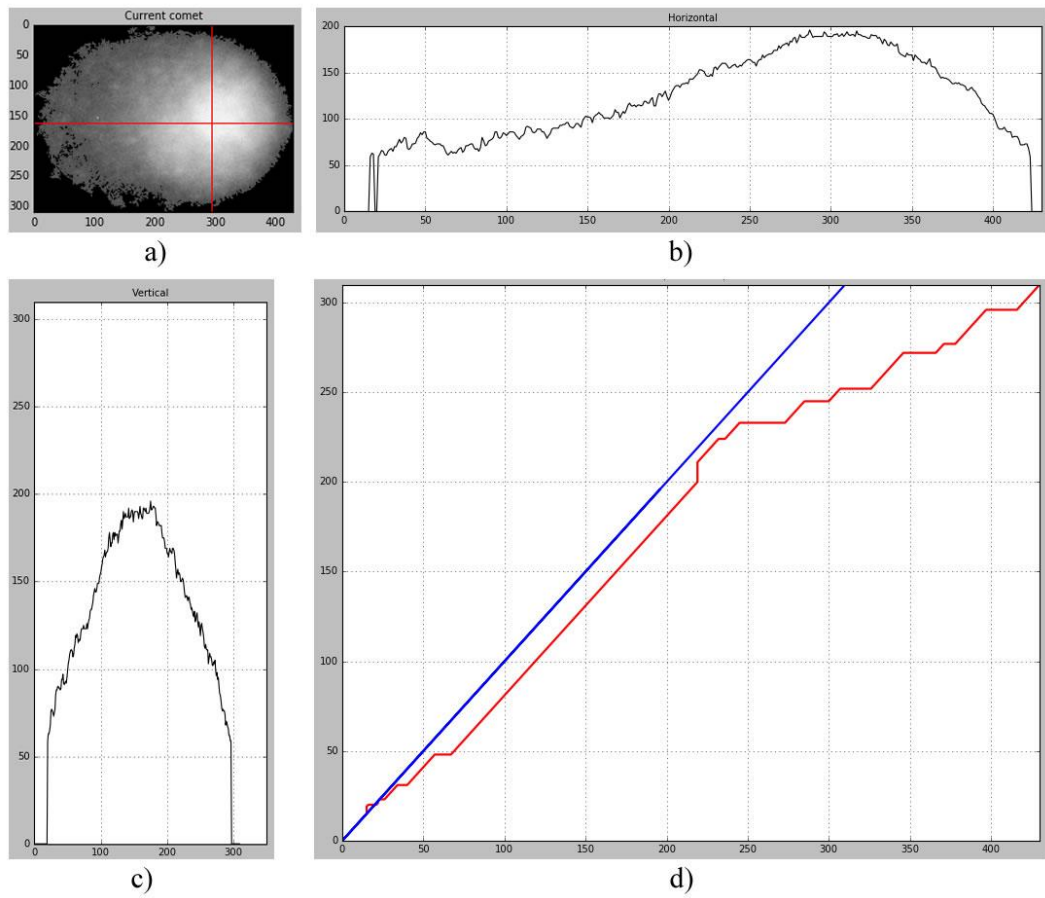


Figure 8.16. G2 comet object. a) Comet image. b) horizontal pixel profile analysis. c) vertical pixel profile analysis. d) diagonal path and path obtained by DTW.

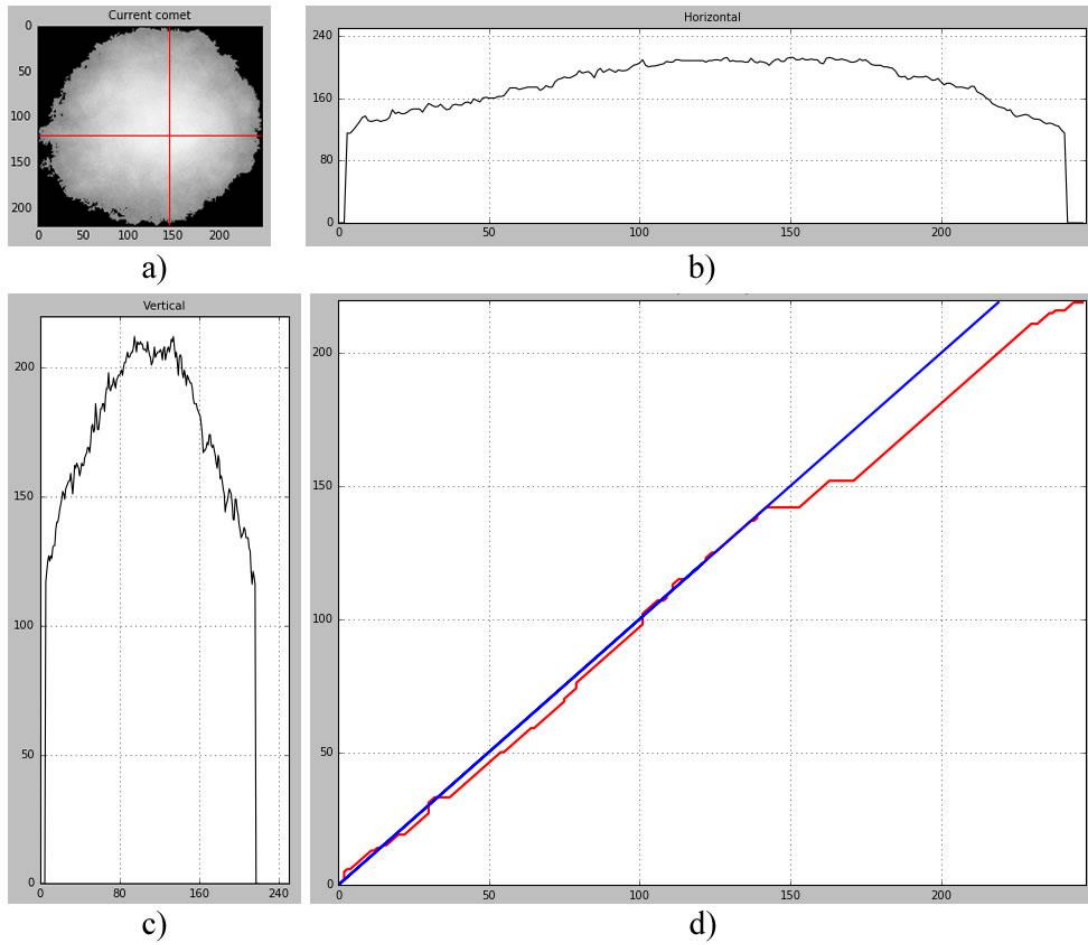


Figure 8.17. G1 comet object. a) Comet image. b) horizontal pixel profile analysis. c) vertical pixel profile analysis. d) diagonal path and path obtained by DTW.

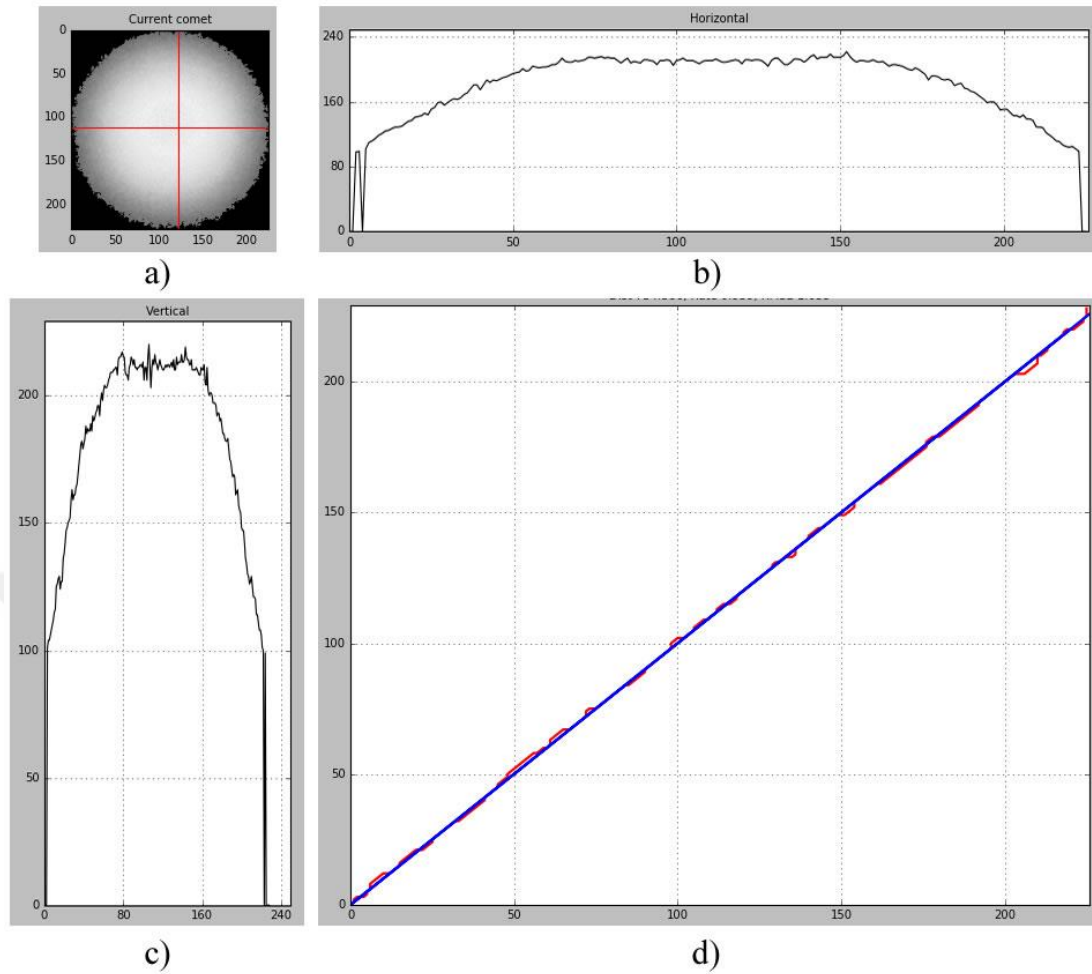


Figure 8.18. G0 comet object. a) Comet image. b) horizontal pixel profile analysis. c) vertical pixel profile analysis. d) diagonal path and path obtained by DTW.

Table 8.4. Measurement parameters of each comet object in Figure 8.15-8.18.

	<b>Figure 8.15</b>	<b>Figure 8.16</b>	<b>Figure 8.17</b>	<b>Figure 8.18</b>
<b>Parameter1</b>	5.736	3.195	7.51	1.79
<b>Parameter2</b>	38933.57	12697.45	9357.3	1829.32
<b>Parameter3</b>	4.245283	2.283688	1.806452	1.135593
<b>Parameter4</b>	0.300767	1.218283	1.087833	3.026202
<b>Parameter5</b>	0.097867	0.643769	1.226656	2.421519
<b>Parameter6</b>	0.097815	0.643655	1.226676	2.421143
<b>Grade</b>	G3	G2	G1	G0



Table 8.5. Interval of six measurement parameters for each damage level.

	G0		G1		G2		G3	
	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.
P <sub>1</sub>	0.004	3.052	1.595	9.351	1.439	10.544	1.873	9.274
P <sub>2</sub>	50.37	3632.41	2256.57	16156.38	6150.18	18978.89	4469.64	60700.0
P <sub>3</sub>	1	1.476563	1.171429	2.727273	1.548837	2.869565	1.023529	6.412698
P <sub>4</sub>	2.484412	22.97448	0.868173	3.221865	0.717342	3.081329	0.177331	3.249537
P <sub>5</sub>	1.101556	20.77371	0.381898	2.57929	0.500317	1.183121	0.037674	2.622324
P <sub>6</sub>	1.101723	20.78649	0.381979	2.579098	0.500375	1.182929	0.037667	2.621876

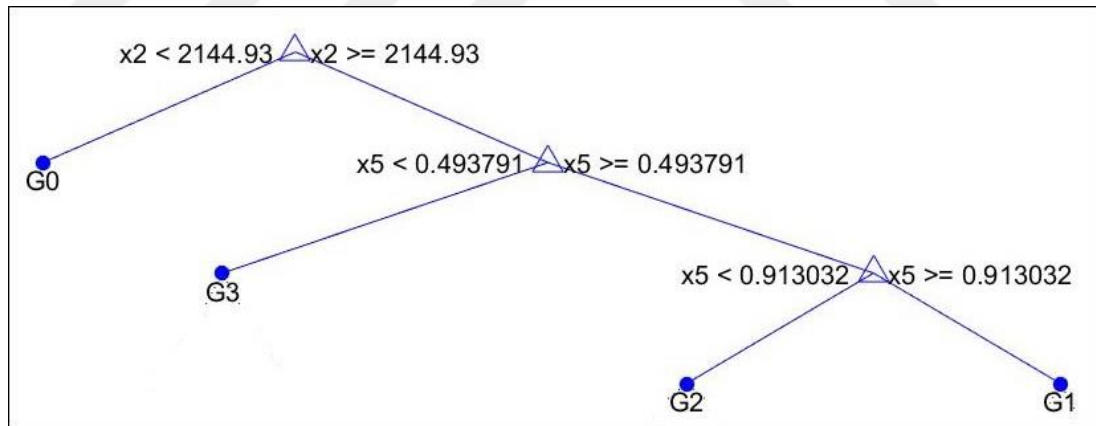


Figure 8.19. Decision tree structure to grade damage level.

When the sample comet object in Figure 8.15 is tested on the decision tree, parameter2 value is firstly checked. Because 38933.57 is greater than 2144.93, it is branched and parameter5 value is checked. Because 0.097867 is lower than 0.493791, this comet object is classified as G3.

When the sample comet object in Figure 8.16 is tested on the decision tree, parameter2 value is firstly checked. Because 12697.45 is greater than 2144.93, it is branched and parameter5 is checked. Because 0.643769 is greater than 0.493791, parameter5 is checked again. Because 0.643769 is lower than 0.913032, this comet object is classified as G2.

When the sample comet object in Figure 8.17 is tested on the decision tree, parameter2 value is firstly checked. Because 9357.3 is greater than 2144.93, it is branched and parameter5 is checked. Because 1.226656 is greater than 0.493791, parameter5 is checked again. Because 1.226656 is greater than 0.913032, this comet object is classified as G1.

When the sample comet object in Figure 8.18 is tested on the decision tree, parameter2 value is firstly checked. Because 1829.32 is lower than 2144.93, this comet object is classified as G0.

Table 8.6. Confusion matrix based on accuracy for each damage level.

<b>Grade</b>	<b>G0</b>	<b>G1</b>	<b>G2</b>	<b>G3</b>	<b>Total</b>
G0	35	0	0	0	35
G1	1	26	2	1	30
G2	0	9	33	3	45
G3	0	1	0	32	33
Total	36	36	35	36	143
Accuracy	99.30%	90.21%	90.21%	96.50%	94.06%

Accuracy of overall is calculated and obtained as below:

$$Accuracy = \frac{35 + 107 + 26 + 103 + 33 + 96 + 32 + 106}{143 \times 4} \times 100 = 94.06\%$$

Table 8.7. Confusion matrix based on sensitivity and specificity for each damage level.

<b>Grade</b>	<b>TP</b>	<b>FP</b>	<b>FN</b>	<b>TN</b>	<b>Sensitivity</b>	<b>Specificity</b>
G0	35	0	1	107	97.22%	100.00%
G1	26	4	10	103	72.22%	96.26%
G2	33	12	2	96	94.29%	88.89%
G3	32	1	4	106	88.89%	99.07%



## CHAPTER 9

### CONCLUSION

In this thesis study, a fully automated desktop application that analyzes and quantifies DNA damage on comet assay images has been developed using Python programming language. The developed application is loaded a comet assay image, makes it complete preprocessing stage, applies a novel thresholding method to obtain binary images, eliminates objects locating at border of loaded image, extracts all individual comet objects, eliminates small objects, eliminates blurry objects, eliminates overlapped comets with a novel method, grades damage level such as G0, G1, G2 or G3 with a novel method, calculates comet parameters and stores results in excel and pdf files under a folder created with loaded image name.

The developed application has some methods that cannot be influenced by nature of comet assay images. The thresholding method, segmentation stage methods, the used method to eliminate overlapped comets, the used method to detect center of head part, the used method to detect tail direction, the used method to separate head part from tail part and the used methods to calculate comet parameters are performed independently on nature of images in a dynamic and non-parametric manner. Even though the thresholding method has a parameter related to loop count, this parameter is suitable for larger and smaller images as well.

The developed application has some methods that can be influenced by nature of comet assay images. Gaussian filter, median filter, the used method to eliminate small objects, the used method to eliminate blurry comet objects and the used method to grade damage level are performed in a parametric manner. Gaussian filter and median filter are not considered to influence substantially. However, kernel sizes of both filters are parametric and may be changed according to size of comet assay images. The used

method to eliminate small objects is influenced because a thresholding parameter has been specified according to mean value of G0 comet areas. The used methods to eliminate blurry objects and grade damage level use decision tree presenting parametric results. With the change nature of images, rules of elimination of blurry comet objects and grading damage level change and wrong elimination and grading occur.

To overcome this problem, decision tree can be developed to find an optimum rule working independently on nature of images. Other machine learning algorithms, artificial neural networks and classification methods can be used.



## REFERENCES

1. Trachootham, D., Lu, W., Ogasawara, M. A., Del Valle, N. R., and Huang, P., "Redox regulation of cell survival", *Antioxidants & Redox Signaling*, 10 (8): 1343-1374 (2008).
2. Gyori, M. B., Venkatachalam, G., Thiagarajan, P. S., Hsu, D., and Clement, M. V., "OpenComet: an automated tool for comet assay image analysis", *Redox Biology*, 2: 457-465 (2014).
3. Halliwell, B., "Why and how should we measure oxidative DNA damage in nutritional studies? How far we come?", *American Journal of Clinical Nutrition*, 72 (5): 1082-1087 (2000).
4. Smolka, B., and Lukac R., "Segmentation of the comet assay images", *Springer-Verlag Berlin Heidelberg*, 3212: 124-131 (2004).
5. Rydberg, B., and Johanson, K. J., "Estimation of DNA strand breaks in single mammalian cells", DNA repair mechanisms 1st ed., Hanawalt, P. C., Friedberg, E. C., Fox, C. F., *Academic Press*, New York, 465-468 (1978).
6. Östling, O., and Johanson, K. J., "Microelectro-phoretic study of radiation-induced DNA damage in individual mammalian cells", *Biochemical and Biophysical Research Communications*, 123 (1): 291-298 (1984).
7. Singh, N.P., McCoy, M.T., Tice, R.R., and Schneider, E.L., "A simple technique for quantitation of low levels of DNA damage in individual cells", *Experimental Cell Research*, 175 (1): 184-191 (1988).
8. Collins, A. R., "The comet assay for DNA damage repair: principles, applications and limitations", *Molecular Biotechnology*, 26 (3): 249-261 (2004).
9. Speit, G., and Rothfuss, A., "The comet assay: a sensitive genotoxicity test for the detection of DNA damage and repair", *Methods Mol Biol.*, 920: 79-90 (2012).
10. Nandkumar, S., Parasuraman, S., Shanmugam, M.M., Rao, K.R., Chand, P., and Bhat, B.V., "Evaluation of DNA damage using single-cell gel electrophoresis (Comet Assay)", *Journal of Pharmacology & Pharmacotherapeutics*, 2 (2): 107-111 (2011).
11. Anderson, D., and Laubenthal, J., "Analysis of DNA damage via single-cell electrophoresis", *Methods Mol. Biol.*, 1054: 209-218 (2013).

12. Collins, A.R., “The comet assay. Principles applications and limitations”, *Methods Mol Biol.*, 203: 163-177 (2002).
13. Olive, P.L., “The comet assay. An overview of techniques”, *Methods Mol Biol.*, 203: 179-194 (2002).
14. Olive, P.L., and Banath, J.P., “The comet assay: a method to measure DNA damage in individual cells.”, *Nat Protocols*, 1 (1): 23-29 (2006).
15. Anderson, D., Dhawan, A., and Laubenthal, J., “The comet assay in human biomonitoring”, *Methods Mol Biol.*, 1044: 347-362 (2013).
16. Collins, A.R., and Azqueta, A., “Single cell gel electrophoresis combined with lesion-specific enzymes to measure oxidative damage to DNA”, *Laboratory Methods in Cell Biology*, 112: 69-92 (2012).
17. Rojas, E., Lopez, M. C., and Valverde, M., “Single cell gel electrophoresis assay: methodology and applications”, *Journal of Chromatography*, 722: 225-254 (1999).
18. Internet: TIOBE (the software quality company), “TIOBE Index for September 2018”, [www.tiobe.com/tiobe-index](http://www.tiobe.com/tiobe-index) (2018).
19. Chityala, R., Pudipeddi, S., “Image Processing and Acquisition using Python 1<sup>st</sup> ed.”, *CRC Press*, Boca Raton, 4-24 (2014).
20. Beazley, D. M., “Python: Essential Reference 4<sup>th</sup> ed.”, *Addison-Wesley Professional*, Boston, 5-25 (2009).
21. Hetland. M. L., “Python Algorithms: Mastering Basic Algorithms in the Python Language 1<sup>st</sup> ed.”, *Apress*, New York, 1-20 (2010).
22. Lutz, M., “Programming Python 3<sup>rd</sup> ed.”, *O'Reilly*, Sebastopol, 40-61 (2006).
23. Vaingast, S., “Beginning Python Visualization: Crafting Visual Transformation Scripts 1<sup>st</sup> ed.”, *Apress*, New York, 31-55 (2009).
24. Internet: NumPy Developers, “NumPy”, [www.numpy.org](http://www.numpy.org) (2018).
25. Blanco-Silva, F. J., “Learning SciPy for Numerical and Scientific Computing 1<sup>st</sup> ed.”, *Packt Publishing*, Birmingham, 5-8 (2013).
26. Internet: SciPy Developers, “SciPy”, [www.scipy.org/scipylib/index.html](http://www.scipy.org/scipylib/index.html) (2018).
27. Internet: SciPy Developers, “Scikits”, [www.scipy.org/scikits.html](http://www.scipy.org/scikits.html) (2018).
28. Internet: Clark, A., “Pillow (PIL Fork)”, [www.pillow.readthedocs.io/en/latest/handbook/overview.html](http://www.pillow.readthedocs.io/en/latest/handbook/overview.html) (2018).

29. Internet: SciPy Developers, “Matplotlib”, [www.matplotlib.org](http://www.matplotlib.org) (2018).
30. Internet: OpenCV dev team, “OpenCV-Python Tutorials”, [www.docs.opencv.org/3.0-beta/doc/py\\_tutorials/py\\_tutorials.html](http://www.docs.opencv.org/3.0-beta/doc/py_tutorials/py_tutorials.html) (2018).
31. Sreelatha, G., Muraleedharan, A., Chand, P., Rajkumar, R. P., and Sathidevi, P. S., “An improved automatic detection of true comets for dna damage analysis”, *Procedia Computer Science*, 46: 135-142 (2015).
32. Sreelatha, G., Rashmi, P., Sathidevi, P. S., Aparma, M., Chand, P., and Rajkumar, R. P., “Automatic detection of comets in silver stained comet assay images for dna damage analysis”, *2014 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC)*, Guilin, 533-538 (2014).
33. Sansone, M., Zeni, O., and Esposito, G., “Automated segmentation of comet assay images using Gaussian filtering and fuzzy clustering”, *Med Biol Eng Comput*, 50 (5): 523-532 (2012).
34. Böcker, W., Rolf, W., Bauch, T., Müller, W., U., and Streffer, C., “Automated comet assay analysis”, *Cytometry*, 35 (2): 134-144 (1999).
35. Vojnovic, B., Barber, P. R., Johnston, P., Gregory, H. C., Marples, B., Joiner, M. C., and Locke, R. J., “A high sensitivity, high throughput, automated single cell gel electrophoresis (‘Comet’) DNA damage assay”, *International Association for Radiation*, Australia, 414-428 (2003).
36. Konca, K., Lankoff, A., Banasik, A., Lisowska, H., Kuszewski, T., Gozdz, S., Koza, Z., and Wojcik, A., “A cross-platform public domain pc image-analysis program for the comet assay”, *Mutation Research*, 534 (1): 15-20 (2003).
37. Rivest, J. F., Tang, M., McLean J., and Johnson F., “Automated measurements of tails in the single cell gel electrophoresis assay”, *Quality Measurements: The Indispensable Bridge between Theory and Reality*, 1: 111-114 (1996).
38. Helma, C and Uhl, M., “A public domain image-analysis program for the single-cell gel-electrophoresis (comet) assay”, *Mutation Research*, 466 (1): 9-15 (2000).
39. Böcker, W., Rolf, W., Bauch, T., Müller, W. U., and Streffer, C., “Technical report: Image analysis of comet assay measurements”, *International Journal of Radiation Biology*, 72 (4): 449-460 (1997).
40. Harrison, C., Pearson, J. D., Bilton, R. F., Burton D. R., and Roberts, J., “The comet moment ratio and other parameters obtained by applying image processing techniques and feature extraction to the SCGE assay”, *Proceedings of the 26th IEEE International Symposium on Computer-Based Medical Systems*, Lubbock, 234-239 (1998).

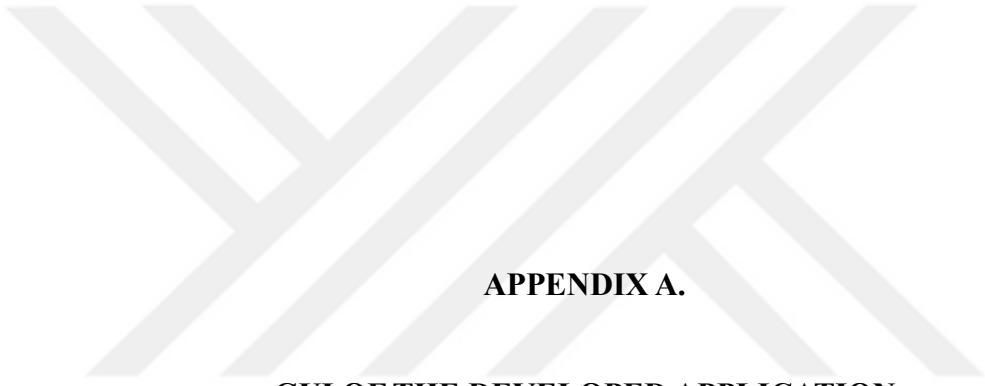


41. Gonzalez, J. E., Romero, I., Barquinero J. E., and Garcia, O., “Automatic analysis of silver-stained comets by CellProfiler software”, *Mutation Research*, 748 (1): 60-64 (2012).
42. Sreelatha, G., Muraleedharan, A., Sathidevi, P. S., Chand, P., and Rajkumar, R. P., “CometQ: An automated tool for the detection and quantification of DNA damage using comet assay image analysis”, *Computer Programs and Methods in Biomedicine*, 133: 143-154 (2016).
43. Lee, T., Lee, S., Sim, W. Y., Jung, Y. M., Han, S., Chung, C., Chang, J. J., Min, H., and Yoon, S., “Robust classification of DNA damage patterns in single cell gel electrophoresis”, *35th Annual International Conference of the IEEE EMBS*, Osaka, 3666-3669 (2013).
44. Mani, U., and Manickam, P., “CoMat: An integrated tool for comet assay image analysis”, *Journal of Pharmaceutical Sciences and Research*, 9 (6): 919-925 (2017).
45. Aykut, T. “C++, Java ve C# ile UML ve Dizayn Paternleri 2<sup>nd</sup> ed.”, *Pusula Yayıncılık*, İstanbul, Türkiye, 3 – 89 (2015).
46. Rumbaugh, J., Blaha, M. R., Lorensen, W., Eddy, F., Premerlani, W., “Object oriented modeling and design”, *Prentice-Hall, Inc. Upper Saddle River*, NJ, USA, 112 – 200 (1991).
47. Booch, G., “Object oriented analysis & design with applications 3rd ed.”, *Addison Wesley Longman Publishing Co. Inc.*, Redwood City, CA, USA, 112 – 200 (2004).
48. Jacobson, I., “Object oriented software engineering: A use case driven approach 1st ed.”, *ACM Press, Addison-Wesley Pub.*, New York, USA, 12 – 200 (1992).
49. Internet: Sparx Systems, “Breakthrough modeling and design”, [https://www.sparxsystems.com/enterprise\\_architect\\_user\\_guide](https://www.sparxsystems.com/enterprise_architect_user_guide) (2017).
50. Gonzalez, R. C., and Woods, R. E., “Sayısal görüntü işleme 3rd ed.”, Telatar, Z., Tora, H., Arı, F., and Kalaycıoğlu, A., *Palme Yayıncılık*, Ankara, 161-739 (2014).
51. Haddad, R. A., and Akansu, A. N., “A class of fast gaussian binomial filters for speech and image processing”, *IEEE Transactions on Acoustics, Speech and Signal Processing*, 39 (3): 723-727 (1991).
52. Chang, C. C., Hsiao, J. Y., and Hsieh, C. P., “An adaptive median filter for image denoising”, *IITA '08. Second International Symposium on Intelligent Information Technology Application*, 346-350 (2008).
53. Vyavahare, A., “Connected component based medical image segmentation”, *International Journal of Innovative Research in Electrical Electronics, Instrumentation and Control Engineering*, 2 (8): 1808-1812 (2014).

54. Foltz, M. A., “Connected components in binary images”, *Machine Vision*, 6: 1-15 (1997).
55. Pertuz, S., Puig, D., and Garcia, M. A., “Analysis of focus measure operators for shape from focus”, *Pattern Recognition*, 46 (5): 1415-1432 (2013).
56. Rosenfeld, A., and Kak, A. C., “Digital picture processing 2nd ed.”, *Academic Press*, New York (1982).
57. Pacheco, J. P., Cristobal, G., Martinez, J. C., and Valdivia, J. F., “Diatom autofocusing in bright field microscopy: a comparative study”, *IEEE 15th International Conference on Pattern Recognition*, 314-317 (2000).
58. Smith, S. W., “Moving average filters, in The Scientist and Engineer’s Guide to Digital Signal Processing 2nd ed.”, *USA: California Technical Publishing*, California (1999).
59. Müller, M., “Information retrieval for music and motion, in: Dynamic time warping”, *Springer-Verlag New York Inc.*, Secaucus NJ, 69-84 (2007).
60. Lemire, D., “Faster retrieval with a two-pass dynamic time warping lower bound”, *Pattern Recognition*, 42: 2169-2180, 2009.
61. Upadyhay, A., Shetty, A., Singh, S. K., Siddiqui, Z., “Land use and land cover classification of LISS-III satellite image using KNN and decision tree”, *IEEE 3rd International Conference on Computing for Sustainable Global Development*, 1277-1280 (2016).
62. Esmeir, S., and Markovitch, S., “Anytime learning of decision trees”, *Journal of Machine Learning Research*, 8: 891-933 (2007).
63. Utgoff, P. E., “Incremental induction of decision trees”, *Machine learning*, 4 (2): 161-186 (1989).
64. Russell, S., and Norvig, P., “Learning decision trees”, *Artificial intelligence: A modern approach*, 3rd ed., *Pearson*, New Jersey, 693-706 (2010).
65. Shannon, E. C., “A mathematical theory of communication”, *Bell System Technical Journal*, 27 (3): 379–423 (1948).
66. Hyndman, R. J., and Koehler, A. B., “Another look at measures of forecast accuracy”, *International Journal of Forecasting*, 22 (4): 679-688 (2006).
67. Stehman, S. V., “Selecting and interpreting measures of thematic classification accuracy”, *Remote Sensing of Environment*, 62 (1): 77-89 (1997).
68. Fawcett, T., “An introduction to ROC analysis”, *Pattern Recognition Letters*, 27 (8): 861-874 (2006).

69. Altman, D. G., and Bland, J. M., “Statistics notes: diagnostic tests 1: sensitivity and specificity”, *BMJ*, 308 (6943): 1552 (1994).
70. DeLong, E. R., DeLong, D. M., and Clarke-Pearson, D. L., “Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach”, *Biometrics*, 44 (3): 837-845 (1988).
71. Powers, D. M. W., “Evaluation: From precision and recall and F-Measure to ROC, informedness, markedness & correlation”, *Journal of Machines Learning Technologies*, 2 (1): 37-63 (2011).





**APPENDIX A.**

**GUI OF THE DEVELOPED APPLICATION**

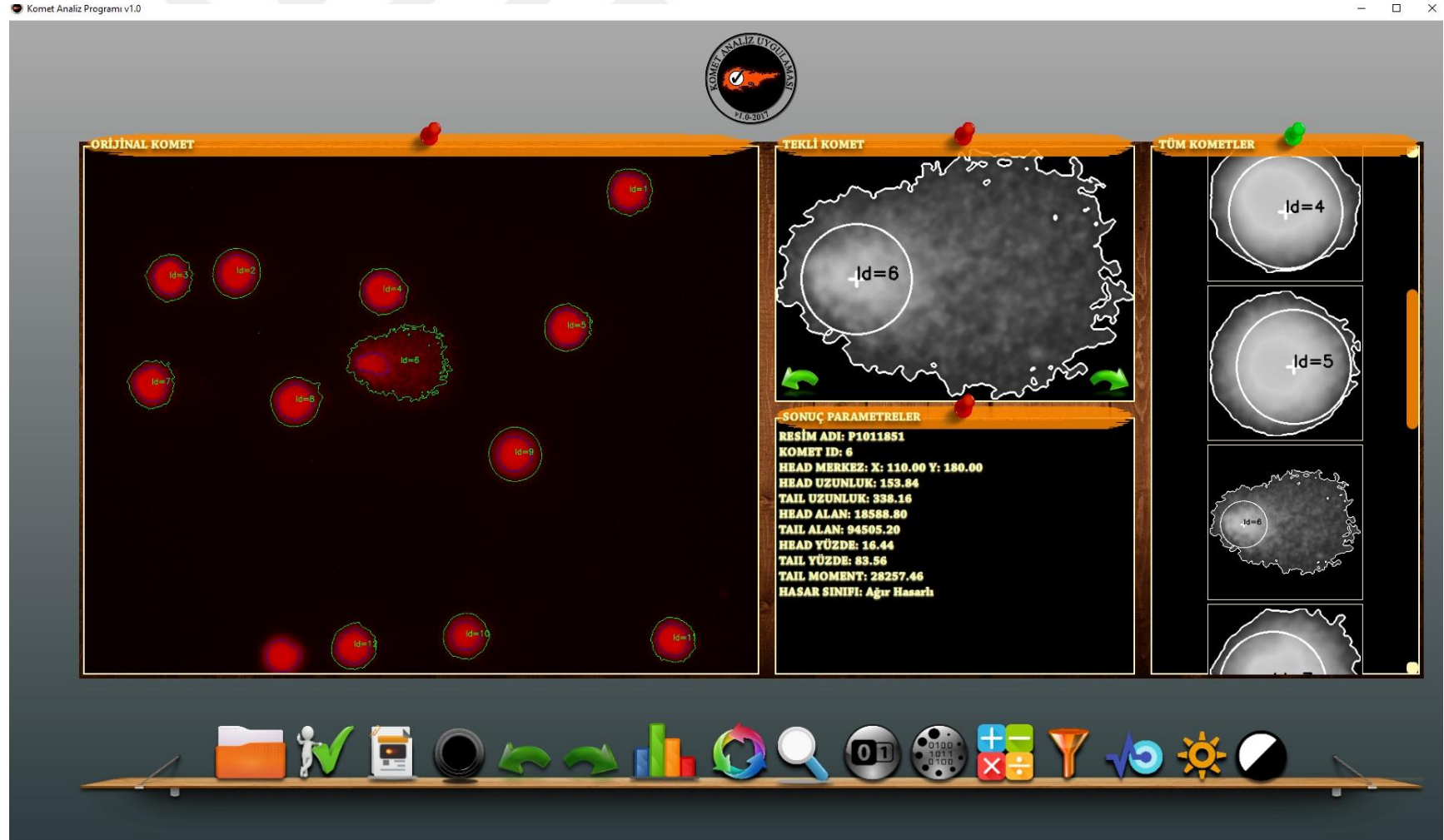


Figure Appendix A.1. GUI of the developed application.

## **RESUME**

Eftâl ŞEHİRLİ was born in Karabük in 1989. He completed primary school education at Safranbolu Ünsal Tülbentçi Primary School in 2003. He completed high school education at Safranbolu Anatolian High School in 2007. He completed bachelor education at Computer Engineering Department of Doğuş University in İstanbul in 2012. He completed master education at Computer Engineering Department of Karabük University in 2014. He made researches for master thesis study as an Erasmus student at Ilmenau Technische Universitat in Germany. He completed PhD education at Computer Engineering Department of Karabük University in 2018. He has worked as a research assistant at Karabük University since 2012.

### **CONTACT INFORMATION**

**Address** : Karabük University, Demir-Çelik Campus,  
Engineering Faculty, Biomedical Engineering Department,  
KARABÜK

**E-mail** : [eftalsehirli@karabuk.edu.tr](mailto:eftalsehirli@karabuk.edu.tr)