Graduate School Form 9
(Revised 3/11)

PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By Albert Parra Pozo

Entitled

An Integrated Mobile System for Gang Graffiti Image Acquisition and Recognition

For the degree of Master of Science in Electrical and Computer Engineering

Is approved by the final examining committee:

1. Chair
2.
3.
4.
5.
6.
7.
8.

Format Approved by:

Chair, Final Examining Committee or **Department Thesis Format Advisor**

☐ **is**

This thesis ☒ **is not to be regarded as confidential.**

Major Professor

To the best of my knowledge and as understood by the student in the *Research Integrity and Copyright Disclaimer (Graduate School Form 20)*, this thesis/dissertation adheres to the provisions of Purdue University's "Policy on Integrity in Research" and the use of copyrighted material.

Major Professor

Approved by: **Head of the Graduate Program** 12/6/11 **Date**

Graduate School Form 20
(Revised 9/10)

PURDUE UNIVERSITY
GRADUATE SCHOOL

Research Integrity and Copyright Disclaimer

Title of Thesis/Dissertation:
An Integrated Mobile System for Gang Graffiti Image Acquisition and Recognition

For the degree of Master of Science in Electrical and Computer Engineering

I certify that in the preparation of this thesis, I have observed the provisions of *Purdue University Executive Memorandum No. C-22,* September 6, 1991, *Policy on Integrity in Research.**

Further, I certify that this work is free of plagiarism and all materials appearing in this thesis/dissertation have been properly quoted and attributed.

I certify that all copyrighted material incorporated into this thesis/dissertation is in compliance with the United States' copyright law and that I have received written permission from the copyright owners for my use of their work, which is beyond the scope of the law. I agree to indemnify and save harmless Purdue University from any and all claims that may be asserted or that may arise from any copyright violation.

Albert Parra Pozo

Printed Name and Signature of Candidate

11/21/2011

Date (month/day/year)

*Located at http://www.purdue.edu/policies/pages/teach_res_outreach/c_22.html

AN INTEGRATED MOBILE SYSTEM FOR GANG GRAFFITI IMAGE ACQUISITION AND RECOGNITION

A Thesis

Submitted to the Faculty

of

Purdue University

by

Albert Parra Pozo

In Partial Fulfillment of the

Requirements for the Degree

of

Master of Science in Electrical and Computer Engineering

December 2011

Purdue University

West Lafayette, Indiana

To those who don't give up.

ACKNOWLEDGMENTS

I would first like to thank my first major advisor, Professor Edward J. Delp, for all the challenges he has given me so far, and for believing that I can overcome them. I really value his advice and criticism; it helps me make the most of my academic career.

I am also very thankful for the support and guidance of my second major advisor, Professor Mireille Boutin. I appreciate her finding the time to help me with the research, and helping me organize my ideas and see things from different points of view.

I would like to thank the remaining member of my Graduate Committee, Professor Charles Bouman, who has indirectly helped me with the color analysis section of my research, as well as other image processing aspects of my project not present on this thesis.

Special thanks to Marc Bosch for his advice and support since we met. He has made my stay at Purdue easier and more enjoyable. Also special thanks to Andrew W. Haddad for his patience and help in my both my academic and personal life.

It has been a pleasure being part of the Video and Image Processing Laboratory (VIPER), both for the quality of the research carried out in the lab and for the people involved. Thanks to my current and former colleagues Dr. Nitin Khanna, Dr. Satyam Srivastava, Dr. Marc Bosch, Andrew W. Haddad, Kevin Lorenz, Aravind Mikkilineni, Dr. Ka Ki Ng, Dr. Fengqing Maggie Zhu, Deen King-Smith, Bin Zhao, Chang Xu, Meilin Yang, and Ye He.

I would like to thank my parents for supporting my career decisions and always believing in me. Thanks to them for giving me the opportunity to acquire and share knowledge with others.

Finally, to Noe for all the understanding and support, and for all the strength she has given me throughout this experience.

The images shown in this thesis were obtained in cooperation with the Indianapolis Metropolitan Police Department. We gratefully acknowledge their cooperation in GARI.

This work was supported by the U.S. Department of Homeland Security's VACCINE Center under Award Number 2009-ST-061-CI000.

TABLE OF CONTENTS

Page

LIST OF TABLES . . . vii
LIST OF FIGURES . . . viii
ABSTRACT . . . xii
1 INTRODUCTION . . . 1
1.1 Proposed System . . . 2
1.2 The Interpretation of Gang Graffiti . . . 5
1.2.1 Some Examples of Interpretation . . . 6
1.2.2 The Use of First Responder Input . . . 10
1.3 Review of Current Methods . . . 13
1.3.1 Graffiti-ID . . . 13
1.3.2 Graffiti Tracker . . . 15
1.3.3 Comparison to GARI . . . 16
1.4 Contributions of This Thesis . . . 18
1.4.1 Publications Resulting From This Work . . . 19
2 COLOR RECOGNITION AND IMAGE SEGMENTATION . . . 20
2.1 Acquisition of Test Images . . . 21
2.2 Color Recognition Based on Touchscreen Tracing . . . 22
2.2.1 Overview of Methods for Color Recognition . . . 23
2.2.2 Color Recognition Using Touchscreen Tracing . . . 23
2.3 Color Image Segmentation Based on Gaussian Thresholding . . . 35
2.3.1 Overview of Methods For Color Image Segmentation . . . 35
2.3.2 Color Segmentation Using Gaussian Thresholding . . . 37
3 SYSTEM IMPLEMENTATION . . . 50
3.1 System Overview . . . 50

Page

3.2 Database of Gang Graffiti 51
3.2.1 Adding Images to the Database 53
3.3 Android Implementation 57
3.3.1 Overview 57
3.3.2 User Interface 58
3.3.3 Security 70
3.4 Web Interface 72
3.4.1 System Overview 72
3.4.2 User Interface 74
3.4.3 Security 83
4 EXPERIMENTAL RESULTS 86
4.1 Execution Time 86
4.1.1 RGB to Y'CH Conversion 87
4.1.2 Database Query Performance 89
4.2 Memory Size 90
4.2.1 Database of Gang Graffiti 91
4.2.2 Android Application 91
5 CONCLUSIONS AND FUTURE WORK 92
5.1 Contributions 92
5.2 Project Status 93
5.3 Future Work 93
LIST OF REFERENCES 95
A IMAGE ACQUISITION PROTOCOL 101
B DATABASE TABLES 108

LIST OF TABLES

Table — Page

1.1 Accuracy and execution time for various numbers of candidate images from the manual annotation matching step. 15

1.2 Comparison of features between Graffiti-ID, Graffiti Tracker, and GARI. 18

2.1 Hue Averages for 100 Test Images (each with graffiti components of different colors). 33

3.1 Main Features of the Android 2.2 Froyo OS. 58

3.2 Main Features of the HTC Desire Mobile Telephone. 59

3.3 Web Browsers Supporting HTML5 Geolocation Service. 73

4.1 Execution Time of the Arithmetic and the Trigonometric Approaches For Color Conversion. 89

4.2 Elapsed Time On the Hand-Held Device and the Server When Uploading an Image. 90

B.1 EXIF data fields in table *images* of the gang graffiti database. 108

B.2 Image location fields in table *images* of the gang graffiti database. . . . 109

B.3 Graffiti analysis fields in table *images* of the gang graffiti database. . . 109

B.4 Image information fields in table *images* of the gang graffiti database. . 110

B.5 User information fields in table *users* of the gang graffiti database. . . . 110

B.6 Image blobs information fields in table *imageBlobs* of the gang graffiti database. 111

LIST OF FIGURES

Figure — Page

1.1 Block Diagram of the GARI System. 3

1.2 Modular Components of the GARI System. 4

1.3 Block Diagram of the Graffiti Analysis Implemented in the Android Application. 4

1.4 Examples of Graffiti Elements. 7

1.5 Examples of Graffiti Color Recognition. 8

1.6 Examples of Manual Recognition of Graffiti Elements. 11

1.7 Example of manual recognition when a crown has been found in a graffiti. The filtering suggests the meaning of this particular element is "a threat against Latin Kings." 13

1.8 Block Diagram of the Graffiti-ID System. 14

2.1 Scaled Color Checkerboard Fiducial Mark. 22

2.2 Scaled USAF 1951 Resolving Power Test Target. 22

2.3 Color Recognition Using Touch Screen Tracing. 24

2.4 Steps For Transforming from RGB to Y'CH Using The Arithmetic Approach. 26

2.5 Warping of the Hexagon Projection Into A Circumference in Our Y'CH Color Space. 27

2.6 3D view of Our Y'CH Color Space (Using the Arithmetic Approach). . 28

2.7 Cross-Section of Constant Hue $H = 0$ rad in Our Y'CH Color Space. . 29

2.8 Cross-Section of Constant Hue $H = \frac{\pi}{3}$ rad in Our Y'CH Color Space. . 29

2.9 Cross-Section of Constant Hue $H = \frac{2\pi}{3}$ rad in Our Y'CH Color Space. . 30

2.10 Bottom View of Our Y'CH Color Space (Using the Arithmetic Approach). 30

2.11 Bottom View of Our Y'CH Color Space (Using the Trigonometric Approach). 31

Figure Page

2.12 Locations of the Average Hue on a bottom view of our Y'CH Color Space (using the trigonometric approach). 33

2.13 Y'CH Color Space Regions Defined by Equation 2.8. 34

2.14 Diagram of Color Image Segmentation Using Gaussian Thresholding. . 38

2.15 Gaussian Thresholding on Red Text, with $\tilde{H} = 0.49$ and $\sigma^2_{\tilde{H}} = 0.05$. . . 41

2.16 Gaussian thresholding on Red Text, with $\tilde{H} = 0.49$ and $\sigma^2_{\tilde{H}} = 0.05$. . . 42

2.17 Gaussian thresholding on White Text, with $\tilde{Y} = 0.83$ and $\sigma^2_{\tilde{Y}} = 0.003$. . 43

2.18 Gaussian thresholding on Black Text, with $\tilde{Y} = 0.13$ and $\sigma^2_{\tilde{Y}} = 0.001$. . 44

2.19 Gaussian thresholding on Blue Text, with $\tilde{H} = 4.01$ and $\sigma^2_{\tilde{H}} = 0.034$. . 45

2.20 Gaussian thresholding on Blue Text, with $\tilde{H} = 4.02$ and $\sigma^2_{\tilde{H}} = 0.020$. . 46

2.21 Gaussian thresholding on Blue Text, with $\tilde{H} = 3.92$ and $\sigma^2_{\tilde{H}} = 0.049$. . 47

2.22 Gaussian thresholding on Black Text, with $\tilde{Y} = 0.17$ and $\sigma^2_{\tilde{Y}} = 0.008$. . 48

2.23 Gaussian thresholding on Black Text, with $\tilde{Y} = 0.19$ and $\sigma^2_{\tilde{Y}} = 0.002$. . 49

3.1 Overview of the GARI System - Client-Side Components (green) and Server-Side Components (blue). 51

3.2 Database Schema Showing the Associations Between the Tables in the Database. 53

3.3 Example of Graffiti (manually labeled). 55

3.4 Database Fields with Information from the Graffiti in Figure 3.3. . . . 56

3.5 Overview of the GARI System. 58

3.6 Difference Between Hardware and Software Keys on the HTC Desire Mobile Telephone. 60

3.7 Main Screen of the GARI Application (most of the layouts support both orientations). 60

3.8 Image Capture. 61

3.9 Progress Dialog - Location Retrieval. 61

3.10 Dialog Indicating Locations Services are Not Enabled. 62

3.11 Dialog Box Notifying the User that No Location Can Be Found. 62

3.12 Crop dialog. 63

Figure Page

3.13 Image Cropping. 63
3.14 Example of Image Browsing. 64
3.15 Browse by Radius. 64
3.16 Screen notifying the user that graffiti are to be downloaded from the server (3.16a) and that the thumbnail images are being retrieved (3.16b). . . . 65
3.17 Results After Querying the Graffiti Database. 65
3.18 Extended Results After Querying the Graffiti Database. 66
3.19 Graffiti Locations Displayed On A Map (black star in white circle is the current location). 66
3.20 Result of Uploading an Image to the Server. The image can be upload successfully (3.20a), or already uploaded to database (3.20b). 67
3.21 Selecting the Region For Color Analysis. 67
3.22 Tracing a Path On The Image. 68
3.23 Gangs Related to the Traced Color (3.23a) and the Graffiti Images in the Database that Match the Traced Color (3.23b). 68
3.24 Graffiti Image Containing a Black 6-Point Star. 69
3.25 Manual Annotation Steps For Identifying the Black 6-Point Star in Figure 3.24. 69
3.26 User ID Prompt. 70
3.27 The Settings Screen. 72
3.28 Overview of the Web Interface of the GARI System. 73
3.29 Main Page of the Web Interface of GARI. 76
3.30 "Archive" Section of the Web Interface of GARI. 77
3.31 "Browse Database" Section of the Web Interface of GARI. 77
3.32 The Current Location of the User is only Acquired Upon Request. . . . 77
3.33 Results of Browsing The Database. 78
3.34 Example of the Interactive Map When a Single Graffiti is Displayed. . 78
3.35 Example of the Interactive Map When Multiple Graffiti are Displayed. 79
3.36 If "Open in a new window" is Clicked, the Interactive Map Expands to Full Screen. 79

Figure Page

3.37 Example of a Popped Out Balloon in the Interactive Map when a Marker is Clicked. 80

3.38 Extended Information for a Specific Search in the Database. 81

3.39 Preview of an Image Before Uploading It to the Graffiti Database. . . . 82

3.40 After Uploading the Image to the Database, information can be added. 84

3.41 Login Page for Accessing the Gang Graffiti Archive. 85

4.1 Execution Time with Respect to the Number of Data Points for the Arithmetic and the Trigonometric Approaches For Color Conversion. 89

A.1 Top view of the setup environment. 106

A.2 Side view of the setup environment. 106

A.3 Graffiti Information Form. 107

ABSTRACT

Parra Pozo, Albert M.S.E.C.E., Purdue University, December 2011. An Integrated Mobile System for Gang Graffiti Image Acquisition and Recognition. Major Professors: Edward J. Delp and Mireille Boutin.

In this thesis we describe a system that allows first responders to collect and browse images of gang graffiti for the purpose of identifying and tracking gang activity. This system is implemented both as an application for Android hand-held devices and as a web-based interface.

Our system includes the acquisition of gang graffiti images and associated metadata. The data obtained from the device is transferred to a database. We also developed methods to provide basic color feature interpretation of gang graffiti images. We describe a color recognition method for hand-held devices based on touchscreen tracing and a color image segmentation method based on Gaussian thresholding. The segmentation creates a probability map that can be used for shape recognition and analysis. A user can browse the image database by graffiti color, location, date and time, with the use of interactive map projections. Additionally, a user can manually add information to the database, such as gang names, symbols, or general comments.

1. INTRODUCTION

Gangs are a serious threat to public safety throughout the United States. Gang members are continuously migrating from urban cities to suburban areas. They are responsible for an increasing percentage of crime and violence in many communities. According to the National Gang Threat Assessment, approximately one million gang members belonging to more than 20,000 gangs were criminally active within the United States as of September 2008 [1]. Criminal gangs commit as much as 80 percent of the crime in many communities according to law enforcement officials throughout the nation. Street, gang graffiti is their most common way to communicate messages, including challenges, warnings and intimidation to rival gangs. It is, however, an excellent way to track gang affiliation and growth or to obtain membership information.

Law enforcement first responders have the potential for finding and documenting graffiti evidence in real time. However, the number of actions that can be taken while on the streets are limited. If there is an incident, or law enforcement needs to compare information, they have to communicate with the corresponding police department. For example, if a new graffiti is spotted by a first responder in an area, the information that can be obtained in situ is very limited. In the best case scenario, the first responder has expertise in gang graffiti interpretation and carries a camera. The only actions he/she can take are reduced to taking a picture and writing down some basic contextual information.

Our long term goal is to develop a system, based on a mobile device such as mobile telephone, capable of using location-based-services, combined with image analysis, to automatically populate a database of graffiti images with information that can be used by law enforcement to identify, track, and mitigate gang activity. We call this system Gang Graffiti Automatic Recognition and Interpretation or GARI. Our first step towards this goal is a system that includes the ability to acquire images in the field using the camera in a mobile telephone and a networked backend database system that uses the metadata available at the time the image is acquired (geoposition, date and time) along with some basic image analysis functions (e.g. color features). The image and metadata is transmitted from the mobile telephone to a server and compared against the database of graffiti images. The matched results are sent back to the user who can then review the results and provide extra inputs to refine the information concerning this graffiti image.

Apart from being able to send and retrieve multimedia data to the database, the first responder can take advantage of location-based-services. The information in the database of gang graffiti can be queried to extract information based on parameters such as date and time of capture, upload or modification of the graffiti image, or radius from a given location. The data includes not only the images, but information related to it, such as date and time, geoposition, gang, gang member, colors, or symbols.

We have implemented these features both as an application for Android handheld devices and as a web-based interface for any device capable of connecting to the Internet (e.g., desktop/laptop computer, iPhone, Blackberry). By providing first responders with this capability, the process of identifying and tracking gang activity can be more efficient.

1.1 Proposed System

Figure 1.1 illustrates a block diagram of our proposed system. It illustrates the various services available, both on the device (no network connection required) and on the server (network connection required). These services include capturing images of gang graffiti, performing automatic analysis and labeling (such as geoposition, date/time, and other EXIF (Exchangeable Image File Format) [2] data obtained from the image), uploading images to the database of gang graffiti, and querying the database to filter and browse its contents.

Figure 1.2 illustrates the modules of our system. Note that the image analysis module is dashed because it is not a required step. There are two basic operation modes: analysis of a new graffiti image and database browsing. The first mode includes capturing or browsing for an existing image on the hand-held device, extracting metadata information and performing color recognition (see Section 2). The second mode includes contacting the database of gang graffiti and obtaining information about graffiti that match a specific query (see Section 3). The two operation modes can be combined. A user can capture an image, perform color recognition, upload it to the database, and browse different fields in an interactive map from the hand-held device. Note that the web-based interface does not include color recognition.

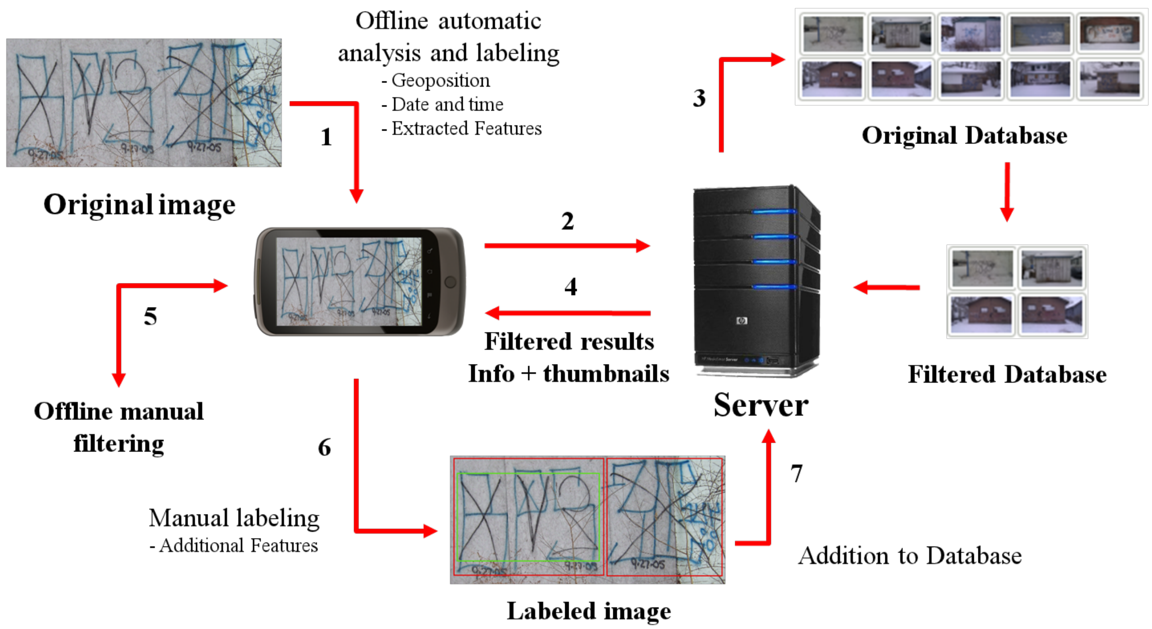


Fig. 1.1.: Block Diagram of the GARI System.

Figure 1.3 illustrates in more detail the submodules of the image analysis module in the Android application. For the web-based interface, only the metadata extraction submodule is used. The metadata extraction submodule includes extracting EXIF data from the image, such as geoposition and date and time, in order to identify the

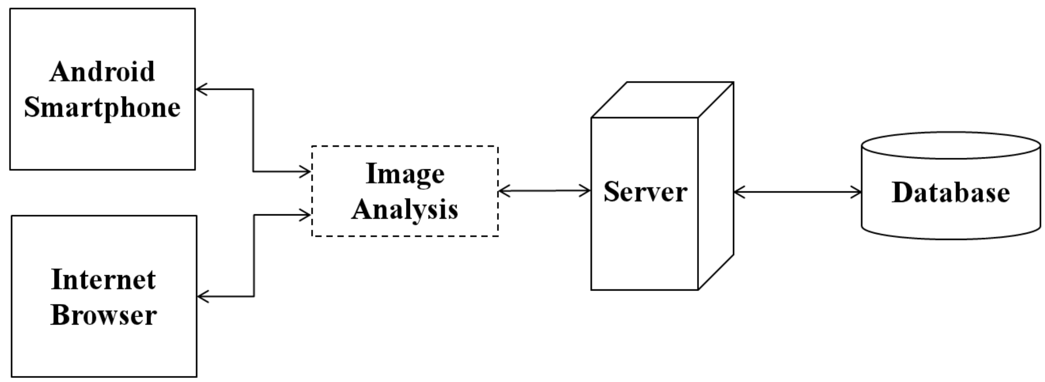


Fig. 1.2.: Modular Components of the GARI System.

image and its location (see Section 3.2). The manual input submodule allows the user to manually identify graffiti components and their properties, such as color and orientation (see Section 3.3.2). The color recognition submodule allows the user to detect the color of a graffiti component by tracing a path using their finger on the device's touchscreen (see Section 2.2.2). The color recognition is done entirely on the device, extra data is obtained for image segmentation from the server, based on color Gaussian thresholding (see Section 2.3).

All the data from the three submodules can be sent to the server along with the graffiti image, and added to the database to be browsed or analyzed in the future.

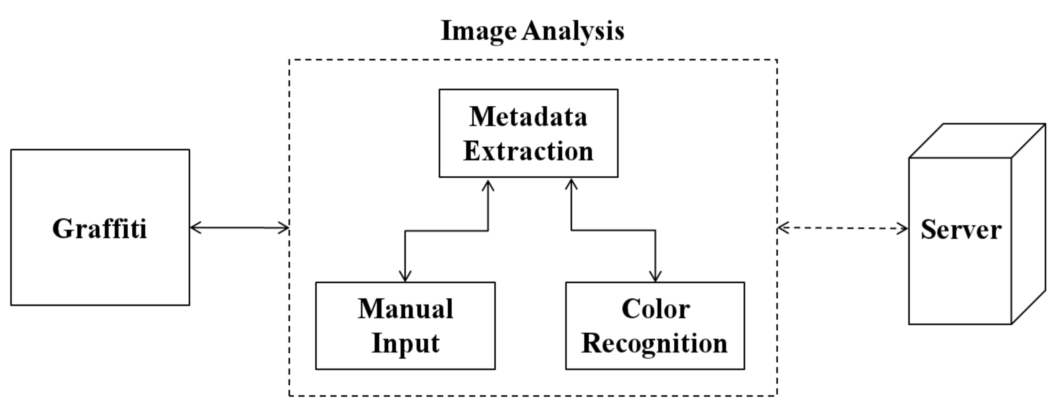


Fig. 1.3.: Block Diagram of the Graffiti Analysis Implemented in the Android Application.

1.2 The Interpretation of Gang Graffiti

Gangs have used street graffiti to communicate with each other for a long time [3, 4]. It is their most common way to communicate messages, including challenges, warnings or intimidation to rival gangs. If graffiti are correctly interpreted, they are a great source of information that can be used to track gang affiliation and growth, or to obtain membership information.

It is worth noting the differences between "graffiti" terms that we use throughout this thesis.

- **Gang**: We use the word gang to refer to a street gang, defined by [5] as a "self-formed association of peers, united by mutual interests, with identifiable leadership and internal organization, who act collectively or as individuals to achieve specific purposes, including the conduct of illegal activity and control of a particular territory, facility, or enterprise".
- **Gang member**: To be distinguished from a tagger. Gang members paint graffiti to mark territory, threaten other gangs or honor other gang members. In contrast, **taggers** paint graffiti to defy authority, or to obtain recognition or notoriety.
- **Gang graffiti**: To be distinguished from tagging. Gang graffiti are simple and usually monochromatic. In contrast, **tags** are artistic and colorful.
- **Component**: Any of the separable elements in a graffiti, such as symbols, acronyms, or numbers.
- **Blob**: Area of the graffiti containing only one component. Useful to identify relative positions of components to each other in the same graffiti.
- **Clique**: Subset of a larger gang with their own name, which may have connection to the gang's neighborhood (e.g., street name, geographic location).

Cliques are local, while gangs extend nationally or internationally. Also known as **factions** or **crews**.

- **Turf**: Slang for territory, or area of influence, specific in this thesis to gangs. Term used when talking about a fight between gangs for territory or power, also known as a turf war, usually with the objective to gain control over the drug market in a specific area.

In the following subsections we describe how to interpret gang graffiti from its contents, including colors, shapes and structure. We also describe how gangs and gang members can be tracked from the graffiti contents and their location. Finally, we illustrate some examples on how a first responder can perform the interpretation and tracking easier and faster. Note we are not claiming in this thesis to be an expert in the interpretation of gang graffiti. Our knowledge is limited. We are relying on law enforcement experts for the GARI project. [1]

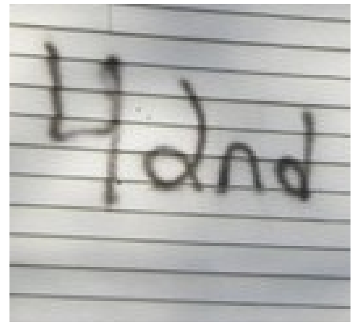
1.2.1 Some Examples of Interpretation

Gang graffiti can be considered a low-level language used by gangs to communicate with each other. The alphabet of this "language" consists not only of letters (Aa-Zz) and numbers (0-9) but also of symbols (e.g., stars, crowns, arrows) and colors. The contents of gang graffiti are simple and straightforward. Gangs usually paint handwritten graffiti using a single color (perhaps two at most). Gang graffiti do not contain complete sentences, but words, short phrases, abbreviations and acronyms (e.g., gang and gang member names, street names and numbers). As is the syntax in a regular language, the relative position and alignment of each component is important in the general structure of the graffiti. The syntax in gang graffiti is two-dimensional. For example, the meaning of a symbol is different if it is painted at the top right of a

[1] The images shown in this thesis were obtained in cooperation with the Indianapolis Metropolitan Police Department (IMPD). We gratefully acknowledge their cooperation in GARI.

graffiti or if the symbol appears upright or upside down. Figures 1.4 and 1.5 illustrate some examples of gang graffiti alphabet, syntax, and color.

(a) Shape



(b) Numbers

(c) Symbols

(d) Colors

(e) Letters

(f) Position and Alignment

Fig. 1.4.: Examples of Graffiti Elements.

(a) Mexicanos Malditos Sureños 13

(b) 18th Street Gang (black) VS Sureños 13 (red)

Fig. 1.5.: Examples of Graffiti Color Recognition.

We will use Figure 1.4 as examples for interpreting gang graffiti. Figure 1.4a is a black gang graffiti. This particular color does not eliminate any gang from being the author of the painting. The 6-point star refers to the Folk Nation, one of the two "nations" to which most gangs belong. Each point means: love, life, loyalty, wisdom, knowledge, understanding. The numbers on both the left and the right of the star, 7 and 4, refer the 7th and 4th letters of the alphabet, G and D, respectively. That is, the Gangster Disciples gang. The three-pointed pitchfork is another sign of the Folk Nation. In this particular case, two upright three-pointed pitchforks make a total of six points, making reference to the 6-pointed star. Moreover, the inscription below the star makes reference to the clique with the street name, *2-8th st* or 28th street, and the nickname of the gang member who painted the graffiti, *Ruthless*.

Figure 1.4b is a black gang graffiti containing the name of a clique, as usual taking its name is taken from the street where they operate. In this case, it refers to the 42nd

Street Gang from Indianapolis. The color itself does not indicate anything concerning which gang this clique may belong to.

Figure 1.4c is a blue gang graffiti with a 6-point star similar to the one in Figure 1.4a. The blue color is used by the Gangster Disciples (and others). The numbers on the sides of the star, along with the additional letters at its bottom make it clear that this graffiti makes reference to the Gangster Disciples. The number 6 in the center of the star is also an extra remainder of the Folk Nation.

Figure 1.4d is a red and black gang graffiti containing the name of a gang/clique in red, Goon Squad (also spelled *Goon Sqaud* or *Goun Sqoud*). This gang/clique name is very common, since it originally refers to a group of thugs or mercenaries associated with violent acts. With the little information from this graffiti it is not possible to determine which gang they belong to or if they are a gang themselves. However, the use of the red color seems to be related to the People Nation, although there are gangs from the Folk Nation that also use the same color. Below the gang name we find the name of the neighborhood where the gang operates (i.e., *Brightwood 2-5st* or Brightwood 25th Street, Indianapolis) in black. The two down arrows at each side of the gang name express turf dominance. The inscription at the very bottom, also in black, appears to be the nickname of the gang member who painted the graffiti, *7MOB*, also known as "Brightwood 7 M.O.B. Bitch." There is an additional down arrow, again expressing turf dominance of this particular gang member.

Figure 1.4d is a simple black gang graffiti containing the acronym *ESG*, referring to the East Street Gang in Indianapolis.

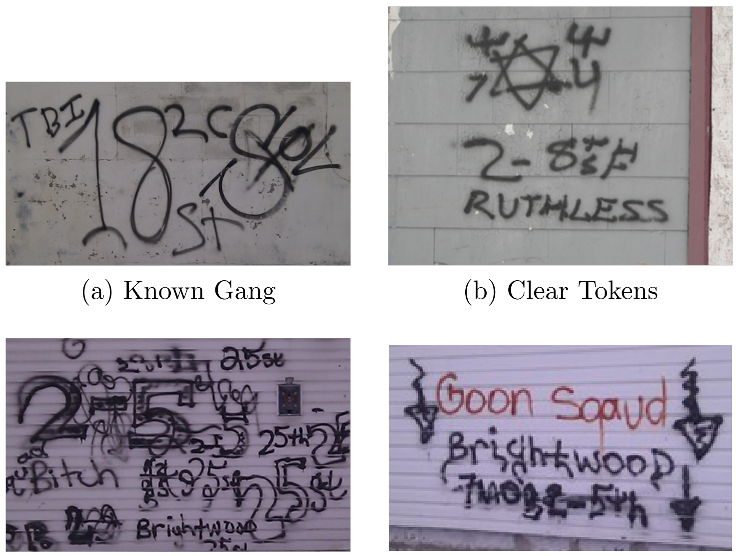
Figure 1.4f is a multicolor gang graffiti. It seems the blue graffiti was painted over the black graffiti. The black graffiti is very similar to the one in Figure 1.4c, belonging to the Gangster Disciples. The 28th Street clique name, along with the nickname *Ruthless*, are also painted next to the 6-point star. The blue graffiti contains the name of a different clique, the *25th Hillside*, from Hillside Avenue in Indianapolis. The inscription at the very bottom, in blue, could make reference to an insult to the gang or gang member who painted the black graffiti originally, however the upside-down 5-point star indicates disrespect for the People Nation. Therefore, both the black and the blue graffiti have been painted by gang members of Folk Nation's gangs, and the blue inscription to the left of the upside-down 5-point star is the nickname of a gang member of the 25th Hillside clique, from the Folk Nation.

1.2.2 The Use of First Responder Input

Automatic image analysis of gang graffiti has some disadvantages. First, depending on the complexity of the analysis, the computational time can be inconvenient especially if the first responder is in the field and would like to know something about a graffiti image they are observing. For example, a first responder may want to detect symbols in the graffiti and browse the graffiti in the database that matches the symbols. This process may involve color and shape segmentation, shape analysis, and finally database querying. Second, the results given by an automated process may not be accurate enough. There are recognition tasks, such as the identification of arbitrary objects, that humans can resolve without effort, but automatic methods are not satisfactory [6]. Color recognition in a scene depends on well-defined illumination and background; shape recognition depends on the position and skewness of the object relative to the camera.

One of the goals of this project is to aid first responders with the process of identification and analysis of gang graffiti. Therefore, our proposed system includes the feature of the user participating in the analysis by manually identifying graffiti content. This allows us to exploit the expertise of the first responder and then use this information in the future to help our methods "learn." It is also a way of providing "groundtruth" information for the automatic analysis.

Figure 1.6 illustrates an example of elements that could be manually filtered from the graffiti. The analysis of the graffiti in each of the subfigures can be improved if the first responder provides some or all the contents of the graffiti. The graffiti in Figure 1.6a can be quickly analyzed if the first responder knows that painting makes reference to the 18th Street Gang. Second, the graffiti in Figure 1.6b contains the nickname of the gang member who painted the graffiti, *Ruthless*. This can be easily identified by just looking at the painting. Third, the graffiti in Figure 1.6c is a more complicated case where the first responder can recognize a font type used by a specific gang or gang member. Finally, the graffiti in Figure 1.6d, already analyzed in Section 1.2.1, contains down arrows, which can be recognized by the first responder. This could then be entered in our database system to improve the types of analysis that can be provided.



(a) Known Gang (b) Clear Tokens

(c) Special Font Type (d) Known Symbols

Fig. 1.6.: Examples of Manual Recognition of Graffiti Elements.

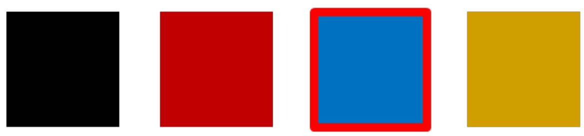
Figure 1.7 illustrates an example of graffiti content filtering. The first responder has taken an image of a gang graffiti containing a crown. The user selects the region where the symbol is painted and tags it as having a crown. The system can then obtain a detailed interpretation through simple questions. First, knowing the symbol contains a crown, the first responder is asked what option is more similar to what is seen (Figure 1.7b). Note that the options include three, five and six point crowns. In this case it is a five-point crown. Second, the first responder is asked about the alignment of the crown (Figure 1.7c). The options in this case are three: upright, crossed upside-down, and upside-down. In this case it is an upside-down crown. Third, the first responder is asked about the color of the crown (Figure 1.7d). In this case it is a blue crown. Lastly, the first responder is asked for additional content that has not been noted along the questions. In this case there are the letters L and K inside the crown. The manual recognition process is complete, and given the answers to the questions a detailed interpretation can be made. First, five-point crowns make reference to the People Nation. However, no conclusions can be drawn from this fact, since given the color, position and alignment of the symbol the meaning can be completely different. Second, upside-down symbols indicate disrespect for the targeted gang by its rivals. It is then a threat to a gang from the People Nation. Third, the L and K letters make reference to the Latin Kings gang. Therefore, this graffiti has been painted by a gang member from any of the Latin Kings rival gangs. The most common Latin King rival gangs are the Spanish Cobras, Maniac Latin Disciples, and any of the Disciples cliques and the gangs from the Folk Nation [7]. However, given that the disrespect is shown through the five-pointed crown, it is more likely to be a graffiti painted by a gang from the Folk Nation. Lastly, the color of the graffiti indicates that the gang member who painted the graffiti is likely to pertain to either the Maniac Latin Disciples or the Gangster Disciples. More details with respect to the other elements of the graffiti (e.g., gang member nickname, gang or clique name, other symbols) are required in order to obtain a more detailed analysis.

Note that a unique result may not be obtainable, because the interpretation depends not only on the symbol itself, but the combination of the contents in the graffiti along with the location and date. The results obtained from the manual recognition are accurate enough so the first responder can browse the matching graffiti and make informed decisions.

(a) Original image

(b) Browse by crown type

(c) Browse by position and alignment



(d) Browse by crown color

Fig. 1.7.: Example of manual recognition when a crown has been found in a graffiti. The filtering suggests the meaning of this particular element is "a threat against Latin Kings."

1.3 Review of Current Methods

There are several methods that have been developed to identify gang graffiti using feature matching as well as tracking gang graffiti using large databases. This section overviews two of the current methods describing their advantages and disadvantages. We also compare both methods with GARI.

1.3.1 Graffiti-ID

Graffiti-ID is an ongoing project (since 2009) at Michigan State University [8]. The project is focused on matching and retrieval of graffiti images. There is nonpublished work which extends the project to gang and moniker identification [9]. The goal of Graffiti-ID is to identify gang/moniker names related to a graffiti image, based on visual and content similarities of graffiti images in a database. Figure 1.8 shows a block diagram of the system. There are two modules, one for populating the database (offline) and another for querying and obtaining results from the database (online). The offline module includes two processes. First, automatic feature extraction using the Scale Invariant Feature Transform (SIFT) [10]. Second, manual annotation of graffiti images by letters and numbers. This is done on images taken from an external gallery of images with the information stored in a database. The online modules includes manual annotation of input images to filter the database and SIFT feature extraction to obtain keypoint matching.

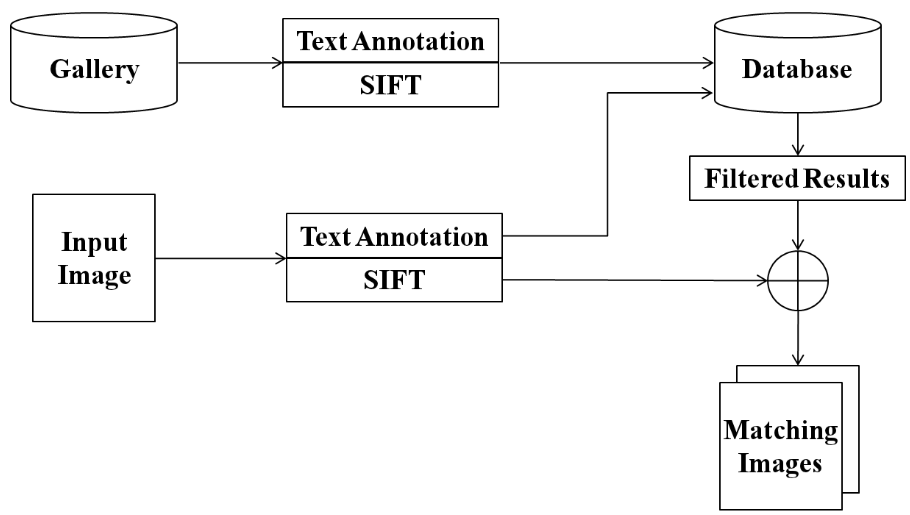


Fig. 1.8.: Block Diagram of the Graffiti-ID System.

The image database used is based on the Tracking Automated and Graffiti Reporting System (TAGRS) from the Orange County Sheriff Department in California. The database consists of 64,000 graffiti images the main sources of the images are the Orange County Transportation Authority and crime reports. A subset of 9,367 images were used for evaluation. Each of these images contains up to four information parameters: moniker, gang, date and time, and address.

The Graffiti-ID system was tested using graffiti images from the original database subset. The retrieval accuracy was evaluated using Cumulative Match Scores [11]. The graffiti images were used as query for the manual annotation matching step, which returns candidate images from the database that match the text description (presence of letters and numbers). SIFT features from the input image and compared against SIFT features from each of the candidate images. The candidates that best match the SIFT features of the query, given the Cumulative Match Scores, are returned to the user. Table 1.1 shows performance results of the output of the second step. The rank-k accuracy refers to the percentage of queries for which the correctly matched images are found within the k candidate images.

Table 1.1: Accuracy and execution time for various numbers of candidate images from the manual annotation matching step.

Candidate Images	**300**	**500**	**1,000**	**9,367**
Rank-30 accuracy	63.8 %	65.4 %	66.5 %	64.3 %
Retrieval Time (seconds/query)	12.4 s	20.1 s	39.8 s	415.7 s

1.3.2 Graffiti Tracker

Graffiti Tracker is a web-based system that began in 2002 [12]. It was designed to help first responders identify, track, prosecute and seek restitution from graffiti vandals. It is primarily used by law enforcement and public works agencies. The database contains more than 2 million manually analyzed graffiti images from 75 cities in two countries and nine states, mainly from the state of California.

The web-based services include graffiti analysis, interactive map browsing, graffiti storing and organization, and graffiti report. Graffiti Tracker provides clients with GPS-enabled digital cameras to generate reports of graffiti activity. The images can then be uploaded through the web interface to the database, they are manually analyzed by trained analysts within 24 hours of submission.

The GPS coordinates of each image are used to build an interactive map where the user can view activity from individual vandals or monikers to specific crews or gangs. Gang trends or migration can be identified if the volume of graffiti for the same gang or vandal is large. A part from the interactive map, the user can browse the stored graffiti by moniker, gang, type of incident, graffiti surface, or removal method. The information can be used to generate reports based on gang or moniker activity, such as total square feet of damage, locations of the incidents, or frequency of graffiti vandalism over a specific period of time.

1.3.3 Comparison to GARI

Although our proposed system, GARI, shares some goals with both of the above systems, our methodology is different. Table 1.2 summarizes a comparison between the features of Graffiti-ID, Graffiti Tracker and GARI.

Both Graffiti-ID and GARI have goals of identifying gangs and gang members based on the graffiti content. Graffiti-ID uses SIFT features between an input image and images from the database. GARI currently uses color recognition techniques, along with metadata information from an image to query the database. Future goals of GARI include the use of SIFT features to detect if an image of a same graffiti was already acquired at a specific location, and also the use of shape techniques to detect graffiti components.

Both Graffiti Tracker and GARI keep track of gang activity based on GPS tags from the images and the graffiti content. However, Graffiti Tracker graffiti analysis is done manually, while GARI's goal is to do most of the analysis automatically.

Graffiti-ID does not exploit the first responders action in the field, such as capture and upload images to a server or browse the database from a mobile device; the analyzed images are on the server. Graffiti Tracker allows users to acquire images only with GPS-enabled cameras they provide and the images have to be transferred to a computer and sent to the server. GARI allows the users to take images with any camera. The GPS coordinates are automatically extracted from the EXIF data of the image or inserted manually when uploaded to the server (i.e., by GPS coordinates or by address - reverse geocoding). Moreover, GARI has a mobile application that allows the user to take an image with a smartphone and send it to the server in situ. GARI also allows the first responder to browse the database of graffiti. GARI allows the user to upload images to the server through a web-based interface from any device capable of connecting to the Internet.

In Graffiti Tracker, image analysis is performed manually by trained analysts with the results obtained within 24 hours of submission. The goal of GARI is to perform the analysis in the field, automatically and in real-time, either on the device or on the server. Graffiti-ID uses SIFT features to match images on the server automatically, but the analysis of the content of the graffiti is done manually, by labeling the image. Moreover, it just allows labels to be numbers (0-9) or letters (a-z), not symbols or other features such as color.

Graffiti-ID does not provide any type of gang activity tracking, while both Graffiti Tracker and GARI provide interactive maps that allows first responders to browse the database and keep track of specific gangs or individuals. The advantage of GARI is that it also provides additional methods for tracking gang activity, including browsing the database by radius from specific locations, or by graffiti color/s. One advantage of Graffiti Tracker is that its database is currently dramatically larger than the GARI database. Therefore, the results retrieved from the Graffiti Tracker database can indicate more accurate gang activity.

In summary, our system combines features from both Graffiti-ID and Graffiti Tracker, and adds more services and functionality. The advantages our system has over Graffiti-ID and Graffiti Tracker are the following. We provide a mobile application that lets first responders act in the field, where the graffiti is located, and upload and browse the database of graffiti in situ. The image acquisition in our system is device independent; any image from any type of camera can be uploaded using either of our supported platforms: an Android-based mobile telephone or through our web-based interface.

Table 1.2: Comparison of features between Graffiti-ID, Graffiti Tracker, and GARI.

Feature	**Graffiti-ID**	**Graffiti Tracker**	**GARI**
Used in field	NO	YES	YES
Graffiti location	Orange County (CA)	9 states (mainly CA)	Indiana
Images in database	64,000	over 2 million	1,418
Analysis (time)	-	Within 24h	minutes (goal)
Analysis (method)	Automatic	Trained analysts	Automatic
Analysis (offline)	Manual labeling	NO	Automatic (goal)
Analysis (online)	Automatic (SIFT)	Manual	Automatic (goal)
Web version	NO	YES	YES
Mobile version	NO	NO	YES
Device	NO	GPS camera	Camera/Smartphone
Interactive Map	NO	YES	YES

1.4 Contributions of This Thesis

In this thesis an integrated mobile system for gang graffiti image acquisition and recognition is described. Our proposed methods include color recognition based on touchscreen tracing and color image segmentation based on Gaussian thresholding. We have also investigated the design and deployment of an integrated image-based database system.

The main contributions in the area of image analysis are as follows:

- We describe a color recognition method based on touchscreen tracing.
- We present a color image segmentation method based on Gaussian thresholding.

The main contributions in the design and deployment of the integrated image-based database system are as follows:

- We developed an integrated image-based database system where data from users and images is connected to gang graffiti information for analysis and tracking.
- We created a web-based interface for first responders and researchers, to upload images and browse gang related information by location, date and time, using interactive maps for better visualization. It is accessible from any device capable of connecting to the Internet, including iPhone and Blackberry.
- We created an Android application for first responders on the field, to upload images to the server, browse gang related information by location using interactive maps, and perform color recognition.

1.4.1 Publications Resulting From This Work

Conference Papers

1. **Albert Parra**, Mireille Boutin, Edward J. Delp, "Location-Aware Gang Graffiti Acquisition and Browsing on a Mobile Device," accepted for presentation at the *Multimedia on Mobile Devices Conference, IS&T/SPIE Electronic Imaging*, San Francisco, CA, January 2012.

2. COLOR RECOGNITION AND IMAGE SEGMENTATION

One of the goals of our system is to identify the color of graffiti components from an image captured by a first responder. For this purpose, we can exploit some restrictions based on general features of gang graffiti. First, gang graffiti content is mostly monochromatic. That is, the graffiti are painted a single color using spray paint, but depending on the distance and the spray pressure one can obtain different tints, tones, and shades of the same color. Second, gang graffiti are always hand-written. Since each graffiti can be written by a different gang member, this excludes the possibility of deciphering writing patterns or styles in most cases. This fact eliminates the use of Optical Character Recognition methods to analyze graffiti contents. Third, gang graffiti are almost always painted on non uniform surfaces with different textures, such as walls, garage doors, or trees. This makes the segmentation of the graffiti contents more challenging. Given these conditions, we decided to acquire gang graffiti images in a systematic way with the cooperation of the Indianapolis Metropolitan Police Department (IMPD). This allows us to work with images so that we can calibrate them with known dimensions in the future.

In our system we developed a method for identifying the color of a graffiti component. We call this approach color recognition based on touchscreen tracing. It runs on a hand-held device without the need of an network connection. We also propose a method for segmenting an image based on the recognized color. We call this color image segmentation based on Gaussian thresholding and for our system it runs on our server. We should note that, due to time constraints, the work reported in this thesis does not use calibrated color images for the color image analysis. We recognize that this does limit the usefulness of our results. We are in the process of further investigation of the use of calibrated color information for the methods we discuss in this chapter.

2.1 Acquisition of Test Images

We need test graffiti images for color and resolution calibration. The image acquisition protocol we followed to properly acquire graffiti images is described in Appendix A. In this protocol we mention the use of a fiducial mark. This mark consists of two elements. The first, shown in Figure 2.1, is a 24x20 inch color checkerboard used to estimate the dimensions and the area of the graffiti and to calibrate the color of the image. The second, shown in Figure 2.2, is a 11x8.5 inch U.S. Air Force (USAF) 1951 Resolving Power Test Target defined by Military Standard 150-A [13]. It is used to estimate the resolving power of the camera used to capture the images. Note that for each graffiti we captured multiple images, using three different cameras and alternating the use of tripods, the color checkerboard and the USAF resolution chart. The methods described in this thesis, including the experimental results, are based on images captured with a tripod and without the use of the color checkerboard or the USAF resolution chart.

We have acquired a large set of these test graffiti images for further work. As we have indicated above we have not used any of the color or resolution calibration information in the studies described in this thesis.

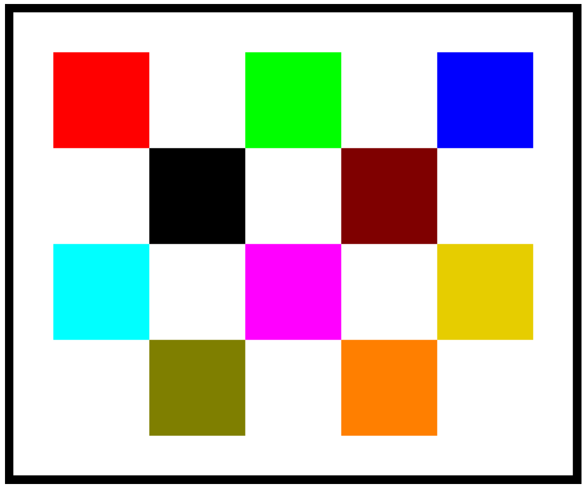


Fig. 2.1.: Scaled Color Checkerboard Fiducial Mark.

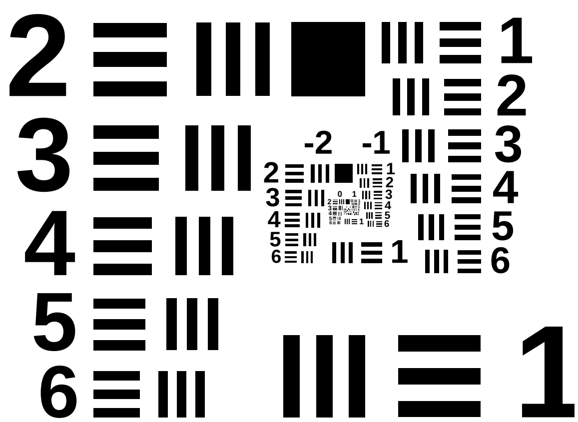


Fig. 2.2.: Scaled USAF 1951 Resolving Power Test Target.

2.2 Color Recognition Based on Touchscreen Tracing

We propose a color recognition method based on user interaction through the Android application of our system. In this method the user traces a color region on the touchscreen display of a graffiti image and then we recognize the color. For this method we use an RGB to Y'CH color space conversion.

2.2.1 Overview of Methods for Color Recognition

Since the first capacitive touchscreen was introduced in 1965 [14] (more detailed description in 1967 [15]) multiple applications have been developed for the use of this device. Some examples include interactive surfaces such as sensitive walls [16], cooperative sharing and exchange of media [17], and freehand manipulation [18].

Most modern mobile devices use touchscreens with tactile feedback to interact with the user. This is used to control the device behavior with gestures [19]. The most common application is the virtual keyboard, which is known to be able to improve the performance of text entry with respect to physical keyboards [20]. However, tactile feedback can also be used to detect a path drawn with the finger on the screen in order to perform image analysis. This technique has been previously used to aid the acquisition of morphometric data from pulmonary tissues [21].

Color recognition techniques are sometimes implemented by using thresholds based on perceptual attributes of specific color spaces. The perceptual thresholds (also known as discrimination thresholds) have been widely studied for human observers [22, 23]. However, some methods do not set the thresholds based on human perceptibility, but based on the application scenario. For example, some skin detection methods use an adaptive skin color filter to detect color regions, by setting thresholds in both RGB and HSV color spaces [24–26].

2.2.2 Color Recognition Using Touchscreen Tracing

Figure 2.3 shows an overview of our color recognition method. Again note that this technique is performed on the hand-held device. First, the user captures an imsge or browses the internal gallery for an image on the device, and draws a path with their finger on the touchscreen. The path is drawn along a component of the graffiti image assumed to have uniform color. Second, the RGB color components of each pixel on the path are converted to a new luma/chroma/hue color space that we call the Y'CH color space. The Y'CH color space is used to intuitively represent color changes in luma or hue in order to obtain the median and the variance of the color along the touch path. Note that we use luma (Y') as opposed to luminance (Y) [27]. Third, we compute three medians on the pixel array that forms the path, namely the luma median ($\tilde{Y}$), the chroma median ($\tilde{C}$) and the hue median ($\tilde{H}$).

We then define three disjoint regions in our Y'CH color space (numbered 3a, 3b and 3c in Figure 2.3), delimited by manually set thresholds based on luma (T_{Y_w}, T_{Y_b}) and chroma (T_C). Depending on which region the medians are located, we do color recognition based on luma (3a) or hue (3b), or no color recognition at all (3c). Finally, we compute a variance around the median based on luma ($\sigma^2_{\tilde{Y}}$) or hue ($\sigma^2_{\tilde{H}}$), if we have decided to perform color recognition based on luma or chroma, respectively. This variance is to be used as an input of the color image segmentation based on Gaussian thresholding (Section 2.3).

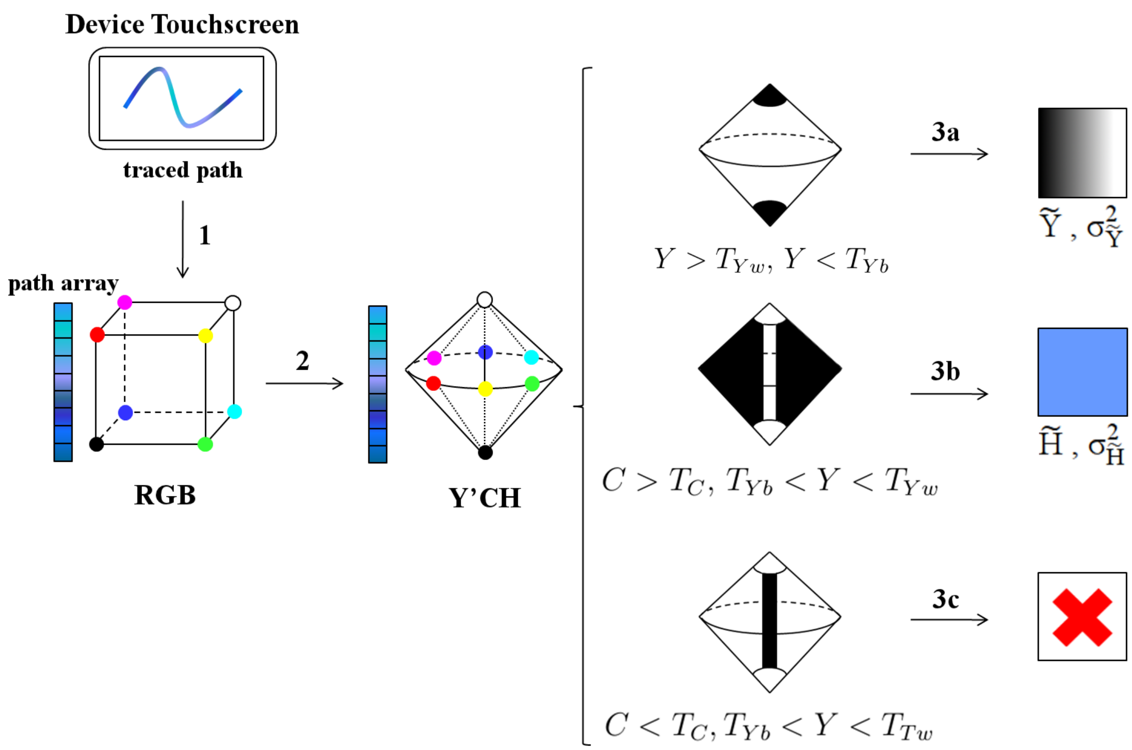


Fig. 2.3.: Color Recognition Using Touch Screen Tracing.

RGB to Y'CH Color Space Conversion

An image captured using our Android application is saved as 32-bit RGB JPEG file, where each pixel is a packed 32-bit integer containing the alpha, R, G and B color components from most to least significant bits respectively. Note that a JPEG image does not have an alpha channel and it is automatically set to zero by the Android bitmap Application Programming Interface (API). From these packed RGB bits we create a three-dimensional array to store the R, G and B components in their unpacked bit representations.

The RGB color space is psychologically non-intuitive because humans have problems with the visualization of a color defined in RGB [28]. The attributes of hue and saturation are the most natural way for humans to perceive colors [29]. The separation of the luma component from the chrominance information is advantageous in image processing. Therefore, we chose to transform the pixels in the image from the RGB color space to our new HSL-based color space, which we call the Y'CH color space, where we carefully define the three dimensions as luma, chroma and hue. We choose chroma over saturation because it better represents human perception of the variation in color purity with respect to luma. In the literature, saturation is defined as relative chroma [30, 31], and the difference must be taken into consideration. For example, the HSL color space is symmetrical with respect to luma, taking the shape of a cylinder. When using chroma the cylinder gets narrower as we move from the center of the neutral axis, forming a shape similar to a bicone [30]. Note that Figures 2.3 and 2.14 illustrate the Y'CH color space solid representation as a bicone for simplicity. However, its true shape is shown in Figure 2.6, where not all the primaries lie in the same plane.

We can convert from RGB to our Y'CH in many ways. In this section we describe two approaches. The first one uses just arithmetic operations, while the second also uses trigonometric operations. We conclude in Section 4 that the first approach is asymptotically faster and hence it is the method that we implemented in our Android application described in Section 3.3.

Our first approach for transforming from RGB to Y'CH, which we call the *arithmetic approach*, is illustrated in Figure 2.4. First, we interpret the RGB cube as being tilted so that the black and white vertices are positioned at the top and the bottom of the neutral axis (vertical axis), respectively. Second, we project the tilted cube onto a plane perpendicular to the neutral axis, thus forming a hexagon. The chroma (C) and hue (H) components in our model are defined with respect to this hexagonal projection (Figure 2.5). Chroma is the distance from the origin of the hexagon to its edge. We can define it as the difference between the largest and the smallest values of an RGB triplet [32] as shown in Equation 2.1. Hue is the angle that represents the angular distance from the red edge of the projection (i.e., set to zero radians) to a particular RGB projection [33, 34], as shown in Equation 2.2. Note that this theoretical hue, which we define as H', is undefined for projections onto the neutral axis (i.e., $C = 0$). Also note that these definitions of chroma and hue correspond to a geometric warping of the hexagon into a circumference.

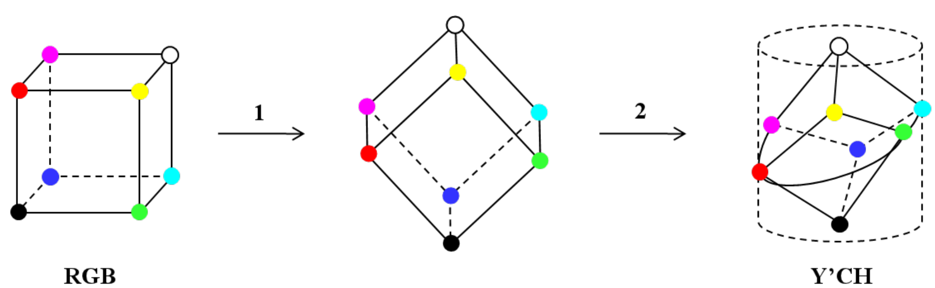


Fig. 2.4.: Steps For Transforming from RGB to Y'CH Using The Arithmetic Approach.

H' is then converted to degrees, which we define as H, by multiplying by 60. This multiplication accounts for $\frac{360°}{6}$, which can be interpreted as the hexagonal analogue of the unit circumference conversion from radians to degrees. That is, since 2π is the perimeter of the unit circumference, we define the conversion as $rad = \frac{360}{2\pi} \times deg$. Since 6 is the perimeter of the unit hexagon, we can define $rad = \frac{360}{6} \times deg = 60 \times deg$.

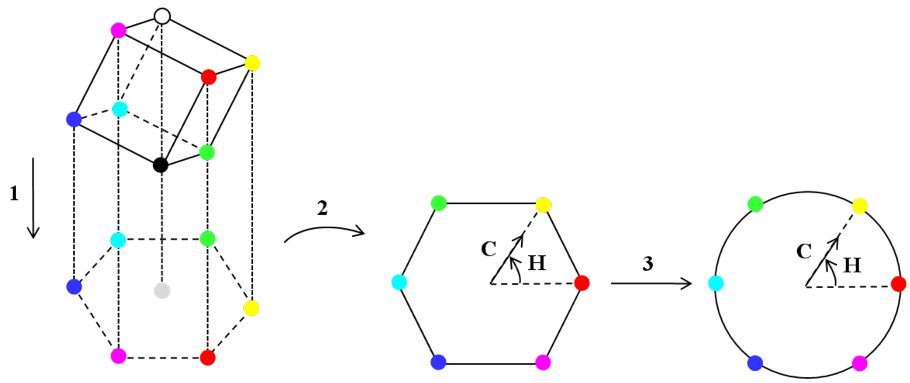


Fig. 2.5.: Warping of the Hexagon Projection Into A Circumference in Our Y'CH Color Space.

Note that we define $H = 0$ when $C = 0$ in order to deal with the undefined hue angle for vector of magnitude zero.

Finally, our luma (Y') is the weighted average of gamma-corrected RGB color components. We define it using the Rec. 601 NTSC primaries [35], as shown in Equation 2.3.

$$\begin{aligned} C &= max(R, G, B) - min(R, G, B) \\ &= M - m. \end{aligned} \tag{2.1}$$

$$H' = \begin{cases} \frac{G-B}{C} & \text{if M=R} \\ \frac{B-R}{C} + 2 & \text{if M=G} \\ \frac{R-G}{C} + 4 & \text{if M=B} \\ \text{undefined} & \text{if C=0} \end{cases} \tag{2.2}$$

$$Y = 0.299R + 0.587G + 0.114B. \tag{2.3}$$

Using these equations, our Y'CH color space is defined in $0 \leq H < 360$ (or $0 \leq H < 2\pi$ in radians), $0 \leq C \leq 1$ and $0 \leq Y \leq 1$. The resulting representation is illustrated in step 3 of Figure 2.4, where each colored dot represents a fully chromatic primary. Given our definitions of luma, chroma and hue, the color space representation does not have a symmetric shape. Figure 2.6 illustrates a 3D view of the Y'CH solid. Figures 2.7 to 2.9 illustrate different cross-sections of constant hue, where the far left and far right corners represent fully chromatic colors. Note that the primaries do not lie in a common luma plane. Also note in Figure 2.8 the effect of setting $H = 0$ where $C = 0$, instead of being undefined. The neutral axis ($C = 0$) does not contain luma values, since the cross-section is not located at $H = 0$. Figure 2.7, however, since it is located at $H = 0$, we do not see any discontinuity.

Figure 2.10 illustrates the bottom view of our Y'CH color space representation, where the hue of different primaries can be identified.

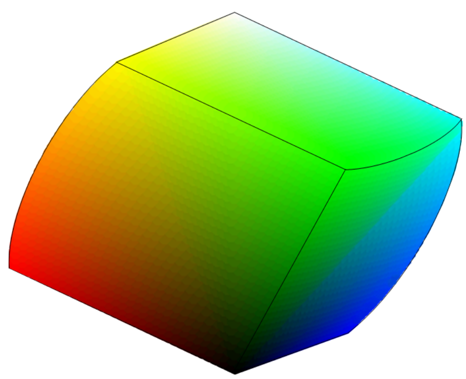


Fig. 2.6.: 3D view of Our Y'CH Color Space (Using the Arithmetic Approach).

Our second approach for transforming from RGB to Y'CH, which we call the *trigonometric approach*, consists of defining the Y'CH color space using cylindrical coordinates, thus skipping the hexagon warping. First, we convert from RGB to Y'IQ using a linear transformation of the RGB cube [36], as shown in Equation 2.4. With this conversion we directly obtain the Y'CH luma, which is defined again using the Rec. 601 NTSC primaries. Then, we can derive the hue and the chroma from a

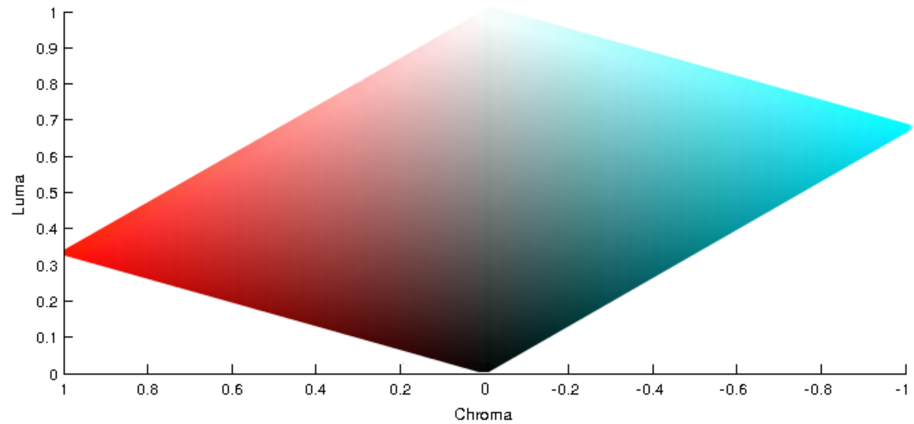


Fig. 2.7.: Cross-Section of Constant Hue $H = 0$ rad in Our Y'CH Color Space.

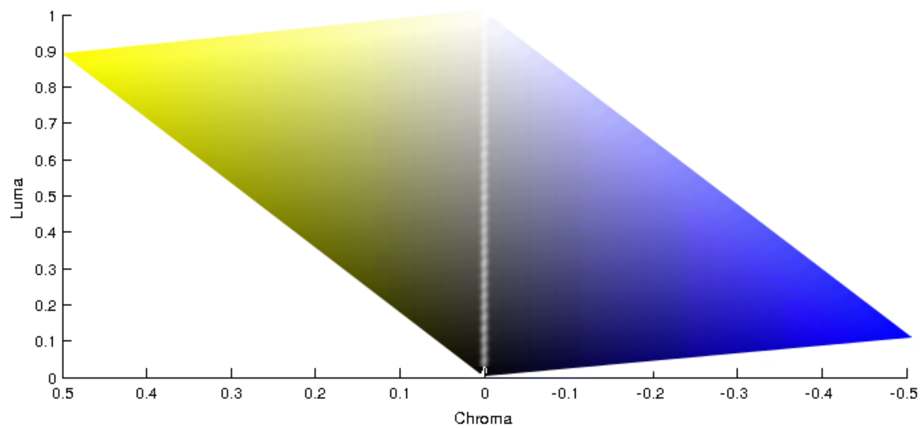


Fig. 2.8.: Cross-Section of Constant Hue $H = \frac{\pi}{3}$ rad in Our Y'CH Color Space.

cylindrical transformation of I and Q [32] as shown in Equation 2.6. Note that the function *atan2* in Equation 2.6 is the two-argument arctangent, defined in Equation 2.7.

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.595716 & -0.274453 & -0.321263 \\ 0.211456 & -0.522591 & 0.311135 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}. \tag{2.4}$$

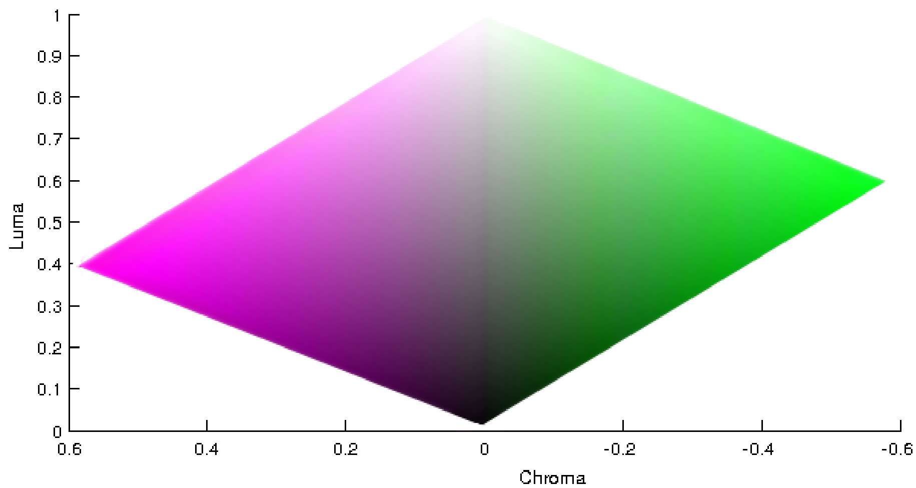


Fig. 2.9.: Cross-Section of Constant Hue $H = \frac{2\pi}{3}$ rad in Our Y'CH Color Space.

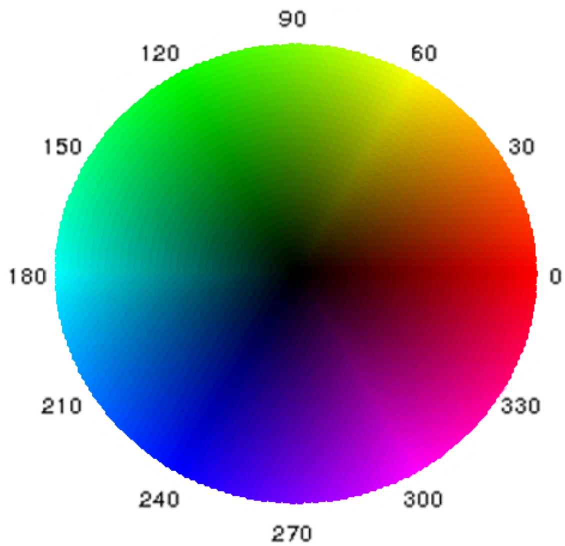


Fig. 2.10.: Bottom View of Our Y'CH Color Space (Using the Arithmetic Approach).

$$H = atan2(Q, I) \tag{2.5}$$

$$C = \sqrt{I^2 + Q^2}, \tag{2.6}$$

$$atan2(I,Q) = \begin{cases} \arctan(\frac{Q}{I}) & I > 0 \\ \pi + \arctan(\frac{Q}{I}) & Q \geq 0,\ I < 0 \\ -\pi + \arctan(\frac{Q}{I}) & Q < 0,\ I < 0 \\ \frac{\pi}{2} & Q > 0,\ I = 0 \\ -\frac{\pi}{2} & Q < 0,\ I = 0 \\ \text{undefined} & Q = 0,\ I = 0 \end{cases} \tag{2.7}$$

Figure 2.11 illustrates the bottom view of our Y'CH color space representation where the hue of different primaries can be identified. Note the hexagon shape.

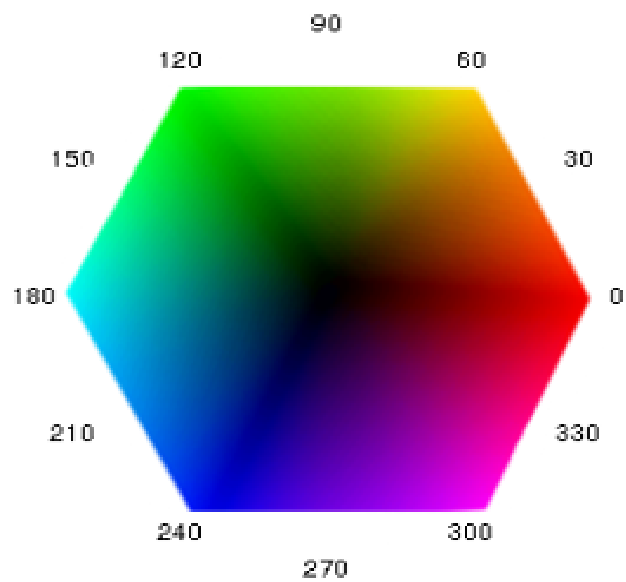


Fig. 2.11.: Bottom View of Our Y'CH Color Space (Using the Trigonometric Approach).

Note that a HSL-based color space, such as Y'CH, has the disadvantage that it does not account for the complexity of the human color perception. However, since we are doing color recognition this is not a problem for us.

Color Median and Variance

Once we have the pixels of the traced path in the Y'CH color space, we want to find which color it represents, and what is the color variance in the Y'CH color space along the path. Note that color is subject to human perception, but in this section it refers to either luma or hue. We can assume that the path has been traced along a surface with approximately uniform color, although we can have some outliers. This will produce a path that has pixels where the color is not from the graffiti component, but from the background. Since the luma or hue of these outliers can be radically different from the rest of the pixels, we find the color median is a better representation than the color mean for this purpose. We compute the median using a selection algorithm known as Randomized-Select [37], which executes in linear time (i.e., with respect to the length of the input pixel array).

We want to distinguish between different colors, including red, blue, black, gold, green, purple and white. The number of colors are empirically chosen based on the contents of 360 graffiti images in our database of gang graffiti. For black and white we focus on the median of the luma component, since they have low chroma and do not have a particular hue.

For red, blue, gold, green and purple, we manually analyzed the hue component of the graffiti in the Y'CH color space from a set of 100 images from our gang graffiti database using the arithmetic approach for the color space conversion. Some of the test images are shown in Figures 2.15 through 2.23. Table 2.1 summarizes the average hues for each color and Figure 2.12 shows the location of these averages on the bottom view of our Y'CH Color Space using the trigonometric approach.

We formed thresholds for luma and chroma to accurately detect the color in the traced path. The thresholds are empirically chosen based on the results of luma and chroma from the test set described above. First, the chroma threshold, defined as T_C in Figure 2.3, is set to 0.05. This threshold accounts for the lowest chroma median among all the traced graffiti components considered to be red, blue, gold, green or

Table 2.1: Hue Averages for 100 Test Images (each with graffiti components of different colors).

Color	Average Hue
Red	6.10 rad
Blue	4.00 rad
Green	2.20 rad
Gold	0.69 rad
Purple	5.15 rad

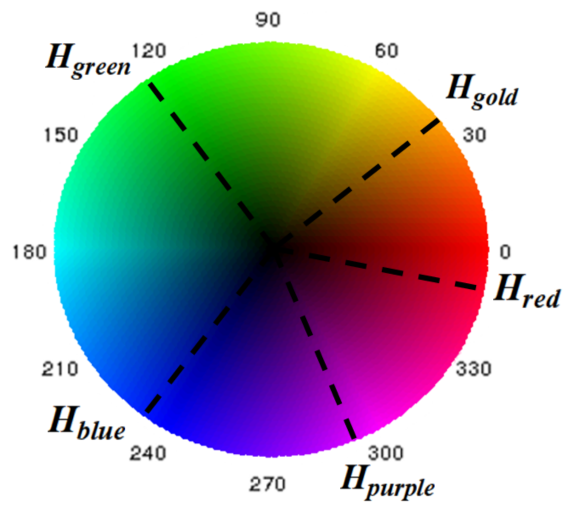


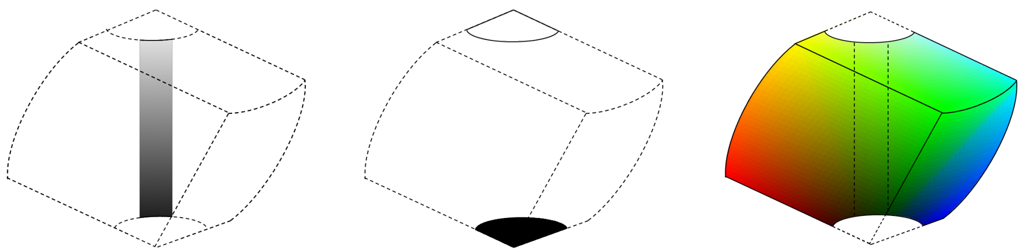
Fig. 2.12.: Locations of the Average Hue on a bottom view of our Y'CH Color Space (using the trigonometric approach).

purple. Second, two luma thresholds are set to 0.12 and 0.85, which we call the black threshold, defined as T_{Yb} in Figure 2.3, and the white threshold, defined as T_{Yw} in Figure 2.3, respectively. These thresholds are for the lowest and highest luma medians among all the traced graffiti components considered to be black or white.

Once we have the pixels of the traced path in the Y'CH color space we use the thresholds. We first compute three medians of the path, based on (1) luma ($\tilde{Y}$), (2) chroma ($\tilde{C}$) and (3) hue ($\tilde{H}$). Depending on the region of the Y'CH color space where the medians are located, we do color recognition of the traced path based on the luma or hue medians. Equation 2.8 defines the different regions, illustrated in

Figure 2.13, and the medians associated with the regions. Note that $\tilde{X}$ represents a generic median.

$$\tilde{X} = \begin{cases} \text{undefined} & \tilde{C} < T_C \ and \ T_{Yb} < \tilde{Y} < T_{Yw} \\ \tilde{Y} & \tilde{Y} < T_{Yb} \ or \ \tilde{Y} > T_{Yw} \\ \tilde{H} & \text{else,} \end{cases} \tag{2.8}$$



(a) Undefined median (b) Luma median ($\tilde{Y}$) (c) Hue median ($\tilde{H}$)

Fig. 2.13.: Y'CH Color Space Regions Defined by Equation 2.8.

Once we have the median, either based on luma or hue, we need to decide which color is associated with it. If the median is based on luma, the color detected is either black or white:

$$color(\tilde{X}) = \begin{cases} \text{black} & \tilde{X} < T_{Yb} \\ \text{white} & \tilde{X} > T_{Yw}. \end{cases} \tag{2.9}$$

If the median is based on hue, we compute the distance between the median and the hues from Table 2.1. We then choose the color associated with the minimum distance. That is, if we define a vector containing the hue averages

$$H_A = \{H_{red}, H_{blue}, H_{green}, H_{gold}, H_{purple}\}, \tag{2.10}$$

the detected color is the minimum distance, H_d, defined as shown in Equation 2.11, where θ is the angular distance between the hue median and an element from the hue average vector.

$$color(\tilde{X}) = H_d = \min_i(\theta(\tilde{H}, H_{A_i})) \tag{2.11}$$

Finally, we compute the variance, $\sigma^2_{\tilde{X}}$, around the median, $\tilde{X}$. This variance is used as an input to the color image segmentation method described next where we use a Gaussian threshold to construct a probability map.

2.3 Color Image Segmentation Based on Gaussian Thresholding

We propose a color image segmentation method based on the results of the color recognition method described in Section 2.2.2. For segmentation we use a Gaussian threshold near a specific luma or hue value in the Y'CH color space, in order to produce a segmented image where each pixel is given a weight depending on its distance from the median.

2.3.1 Overview of Methods For Color Image Segmentation

Since the advent of color imaging most of the image segmentation techniques were proposed for gray-level images [38–41], due to the fact that working with the color components substantially increases the computational complexity of the method [42]. More precisely, for each pixel in a 24-bit quantized image we have $2^{24} = 16,777,216$ distinct colors. There has been a remarkable growth of color image segmentation approaches [43–45]. Image color segmentation techniques can be divided into three categories [45].

- *Physics based techniques*: uses models of physical interaction between the objects to be segmented and their behavior under different lightning conditions.

These models include the dichromatic reflection model [46] and the unichromatic reflection model [47] for single illumination sources, and a more general model of image formation [48] for multiple illuminations.

- *Feature-space based techniques*: uses feature spaces where the spatial relationship between colors is ignored. These techniques are of three types. (1) Clustering of regions given patterns with specific properties, including methods such as *k-means clustering* [49] or *Iterative Self-Organizing Data Analysis Technique (ISODATA)* [50]. These methods are based on classification of color patterns. (2) Adaptive k-means clustering, including methods based on *maximum a posteriori (MAP)* estimation [51] or split-and-merge strategies [52]. These methods are based on the combination of k-means clustering, but some adaptive spatial constrains are used. (3) Histogram thresholding, including methods based on RGB thresholding [53], hue information [54], specific skin color domains [55], or entropy thresholding [56]. These methods are based on the identification of objects in the image by analyzing the maximums and minimums of the color histograms using different color spaces and perceptual attributes.

- *Image-domain based techniques*: uses both feature-space homogeneity and spatial compactness. That is, feature spaces where the spatial relationship between colors is exploited. These techniques can be subcategorized into four groups. (1) Split-and-merge, including methods such as region smoothing by *Markov Random Fields (MRF)* [57] or *Gaussian Markov Random Fields (GMRF)* [58]; classification by k-means [59]; or splitting by either *watershed transform* [60] or quad-tree image representation for segmentation of skin cancers [61], among others. These methods are based on an initial segmentation followed by one or more merging phases in order to connect common structures. (2) Region growing, including methods such as RGB color distribution growing [62], HSV morphological open-close growing [63], or color quantization growing [64]. These methods are based on gradually accumulating pixels from the image satisfying specific criteria forming segmented regions. (3) Neural-network based classification, including methods such as minimization of *Hopfield networks* [65], or background extraction using two three-layered neural network [66]. These methods are based on classification of neural networks, that is, interconnected groups of artificial neurons, each of one performs simple operations. (4) Edge based techniques, including methods such as combination of HSI gradients [67], active contours [68], or the *Mumford-Shah variation model* [69]. These methods are based on region edge detection using color gradient functions.

Our approach falls under the second category, feature-space based techniques. However, it differs from the methods mentioned above. Although there are some techniques in the literature that also work with only hue or luma information, either circular histogram thresholding [54] or one-dimensional histogram thresholding [70], we do not obtain the descriptors of the probability distribution from the color histogram of the image. Instead, the median and the variance obtained during the color recognition process are used for segmentation. Our segmentation approach does not produce binarized images, but gradient images weighed by a Gaussian distribution around the median, thus creating a probability map for a specific luma or hue. These types of probability maps are used for increased accuracy and robustness in some clustering techniques [71, 72].

2.3.2 Color Segmentation Using Gaussian Thresholding

Figure 2.14 shows an overview of our color segmentation method divided in 5 steps. Note that we currently use this method on the server in our system and do not use it on the hand-held device. We assume that, given a graffiti image, we have the median, $\tilde{X}$, and the variance, $\sigma^2_{\tilde{X}}$, of a traced path obtained from the color recognition process we described in Section 2.2.2 (step 1a). We then transform the entire RGB image to the new Y'CH color space (step 2). We segment the image using Gaussian thresholding (steps 3 to 5). This method is based on the luma or hue depending on our manually set thresholds.

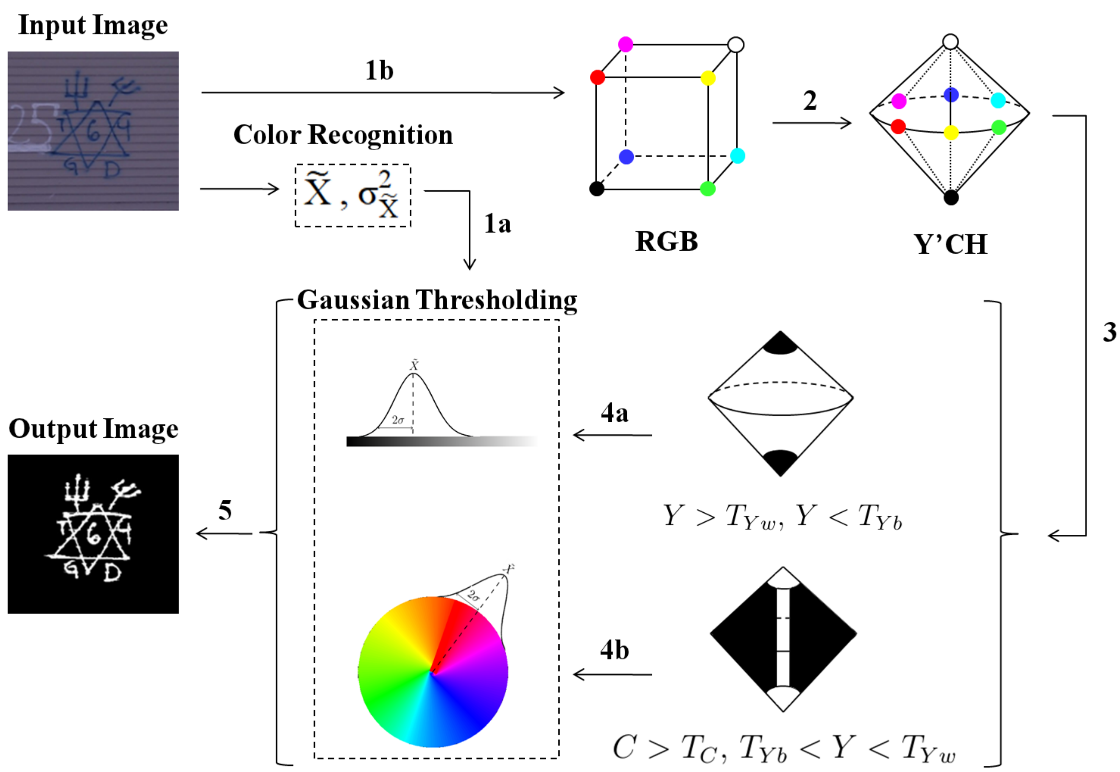


Fig. 2.14.: Diagram of Color Image Segmentation Using Gaussian Thresholding.

The image segmentation process (steps 3 to 5) is as follows. We assume we have already done color recognition on a graffiti component of the input image, so we have the median, $\tilde{X}$ and the variance, $\sigma^2_{\tilde{X}}$, where X corresponds to hue or luma, depending on the results of Equation 2.8 in Section 2.2.2. We scan the Y'CH input image and set to zero all pixels that fall outside the thresholds given in Equation 2.12, where the output $X_t(i,j)$ refers to either the luma (step 4a) or the hue (step 4b) component of an image pixel at a particular location, $X(i,j)$, and $Y(i,j)$ and $C(i,j)$ correspond to its luma and chroma components, respectively. Note that the thresholds T_C, T_{Y_w} and T_{Y_b} are the same as the ones used for color recognition in Section 2.2.2. For the rest of the pixels we use a weight based on a normal distribution, as shown in Equation 2.14, establishing a confidence interval of $2\sigma_{\tilde{X}}$. That is, all pixels outside of two standard deviations from the median $\tilde{X}$ are set to zero, and the pixels inside the confidence interval are given a weighted value (step 5).

$$X_t(i,j) = \begin{cases} 0 & C(i,j) < T_C \textit{ and } T_{Yb} < Y(i,j) < T_{Yw} \\ X(i,j) & \text{else} \end{cases} \tag{2.12}$$

$$X_o(i,j) = \begin{cases} \frac{1}{\sqrt{2\pi\sigma^2_{\tilde{X}}}} e^{-\frac{(X_t(i,j)-\tilde{X})^2}{2\sigma^2_{\tilde{X}}}} & |X_t(i,j)| < 2\sigma_{\tilde{X}} \\ 0 & \text{else} \end{cases} \tag{2.13}$$

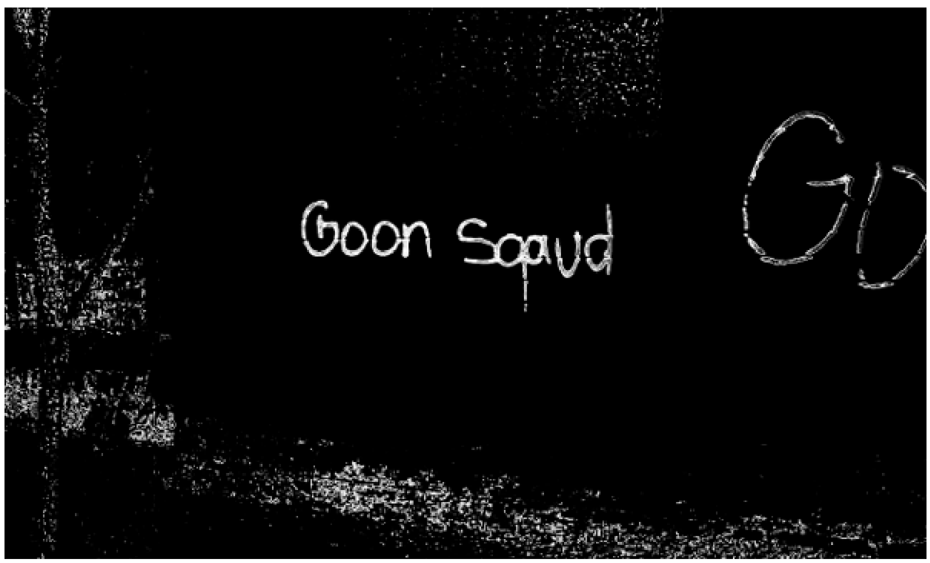
Overall, the steps 3 to 5 in Figure 2.14 can be summarized as

$$X_o(i,j) = \begin{cases} 0 & C(i,j) < T_C \textit{ and } T_{Yb} < Y(i,j) < T_{Yw} \\ \frac{1}{\sqrt{2\pi\sigma^2_{\tilde{X}}}} e^{-\frac{(X(i,j)-\tilde{X})^2}{2\sigma^2_{\tilde{X}}}} & |X(i,j)| < 2\sigma_{\tilde{X}} \\ 0 & \text{else} \end{cases} \tag{2.14}$$

If we use this criteria for all the pixels in the image, we obtain a grayscale image as output, with pixel values in the range $[0, 255]$. Any pixel value greater than zero is part of the detected region, and the higher its value is, the closer it is to the median from the color recognition process. That is, the closer $X(i,j)$ is to $\tilde{X}$, the more similar it is to the graffiti component previously traced. Figure 2.15 shows an example where the color recognition process is performed by tracing a path along the red letters "G" and "o" in the graffiti component "Goon Sqaud" (Figure 2.15a), with thresholds T_{Y_b}, T_{Y_w} and T_C as defined in Section 2.2.2. The output parameters, based on hue, are $\tilde{H} = 0.49$ and $\sigma^2_{\tilde{H}} = 0.05$. Then, in the image segmentation process we use Equations 2.12 and 2.12 on every pixel in the image, resulting in a grayscale image (Figure

2.15b). Figure 2.16 shows the effect of the Gaussian thresholding process on the letters "Go". Note that it does not produce a binary image, and that the pixel values decrease during the transition from segmented component to background. Figures 2.17 to 2.23 illustrate different input images and their outputs. If we normalize the pixel values to the range $[0, 1]$ this method produces a probability map, where higher values are represented by higher intensity values. This map can be useful for shape analysis in the future (see Section 5.3).

(a) Uploaded Image



(b) Segmented Image

Fig. 2.15.: Gaussian Thresholding on Red Text, with $\tilde{H} = 0.49$ and $\sigma^2_{\tilde{H}} = 0.05$.

(a) Uploaded Image (b) Segmented Image

Fig. 2.16.: Gaussian thresholding on Red Text, with $\tilde{H} = 0.49$ and $\sigma^2_{\tilde{H}} = 0.05$.

(a) Uploaded Image

(b) Segmented Image

Fig. 2.17.: Gaussian thresholding on White Text, with $\tilde{Y} = 0.83$ and $\sigma^2_{\tilde{Y}} = 0.003$.

(a) Uploaded Image

(b) Segmented Image

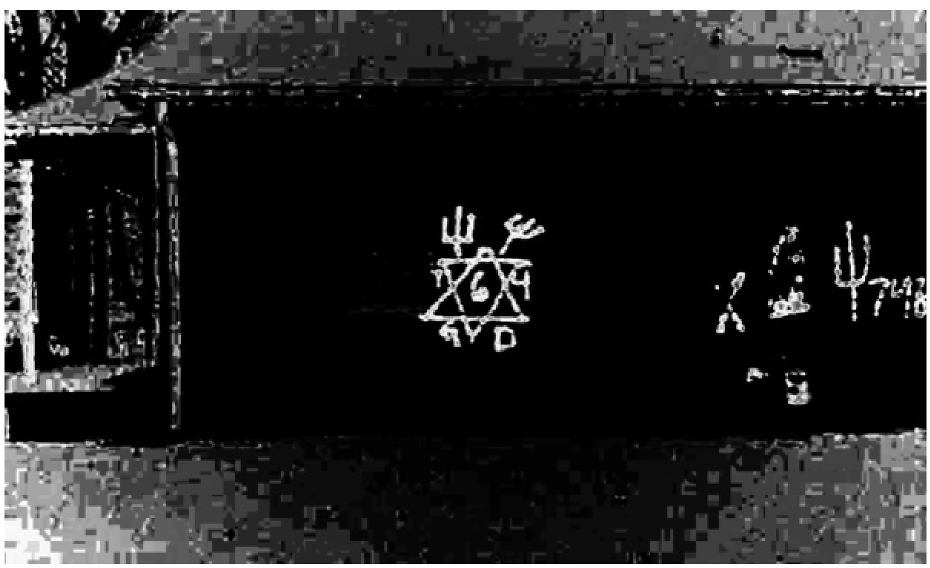
Fig. 2.18.: Gaussian thresholding on Black Text, with $\tilde{Y} = 0.13$ and $\sigma^2_{\tilde{Y}} = 0.001$.

(a) Uploaded Image

(b) Segmented Image

Fig. 2.19.: Gaussian thresholding on Blue Text, with $\tilde{H} = 4.01$ and $\sigma^2_{\tilde{H}} = 0.034$.

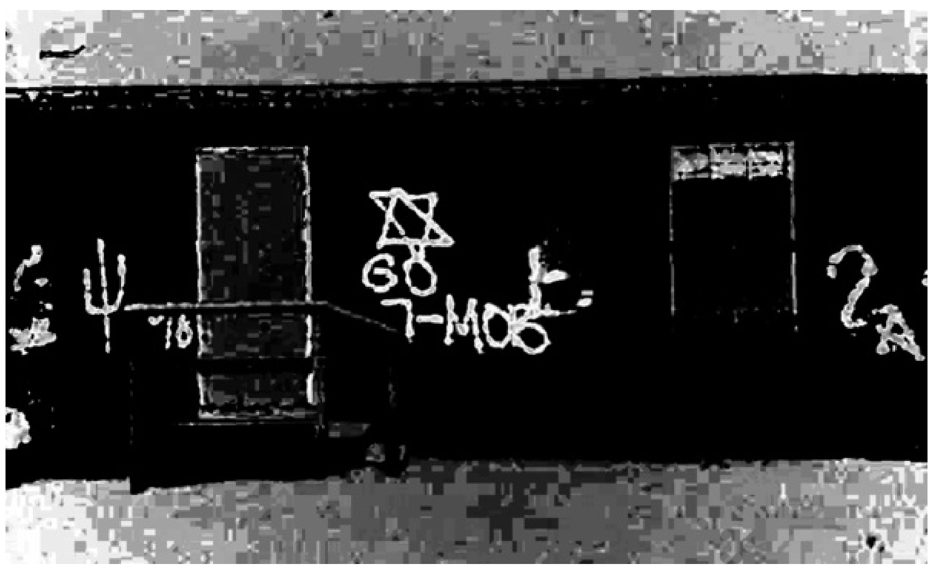
(a) Uploaded Image



(b) Segmented Image

Fig. 2.20.: Gaussian thresholding on Blue Text, with $\tilde{H} = 4.02$ and $\sigma^2_{\tilde{H}} = 0.020$.

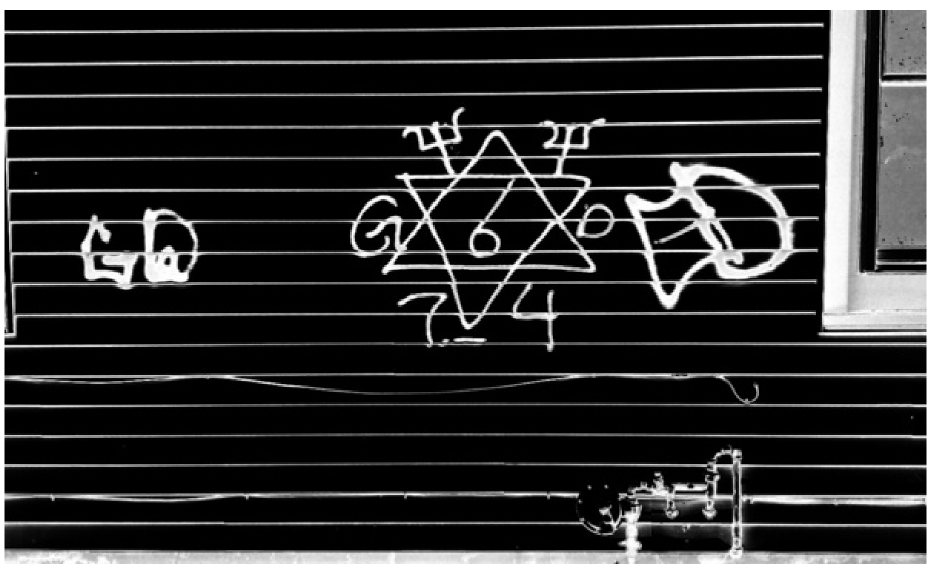
(a) Uploaded Image



(b) Segmented Image

Fig. 2.21.: Gaussian thresholding on Blue Text, with $\tilde{H} = 3.92$ and $\sigma^2_{\tilde{H}} = 0.049$.

(a) Uploaded Image



(b) Segmented Image

Fig. 2.22.: Gaussian thresholding on Black Text, with $\tilde{Y} = 0.17$ and $\sigma^2_{\tilde{Y}} = 0.008$.

(a) Uploaded Image

(b) Segmented Image

Fig. 2.23.: Gaussian thresholding on Black Text, with $\tilde{Y} = 0.19$ and $\sigma^2_{\tilde{Y}} = 0.002$.

3. SYSTEM IMPLEMENTATION

3.1 System Overview

We implemented a prototype of the GARI system as an application for Android devices and as a web-based interface accessible from any web browser. Figure 3.1 illustrates the GARI system, which is divided in two groups:

1. **Client-side**: Perform operations on the Android device and communicate with the database of gang graffiti through either the WiFi or 3G networks.

2. **Server-side**: Perform operations on the database of gang graffiti and communicate with the client.

The client-side includes the device and methods available to the users, either to operate without the use of a network connection (offline services) or to make queries to the database (online services). The offline services are only available from Android devices (Section 3.3). The online services are available from both Android devices or any web browser (e.g., Internet Explorer, Mozilla Firefox, Google Chrome). This includes desktop and laptop computers as well as iPhone and Blackberry smartphones (Section 3.4). However, note that the iPhone currently blocks file uploads from its default web browser (Safari) at this time and hence provides limited functionality. The server-side includes all operations performed on the server including image analysis and queries to the database from both the Android application and the web-based interface. The database comprises gang graffiti images and metadata information for each entry, such as EXIF data, image geolocation and the results of the image analysis on each image whether it was performed on the server or client.

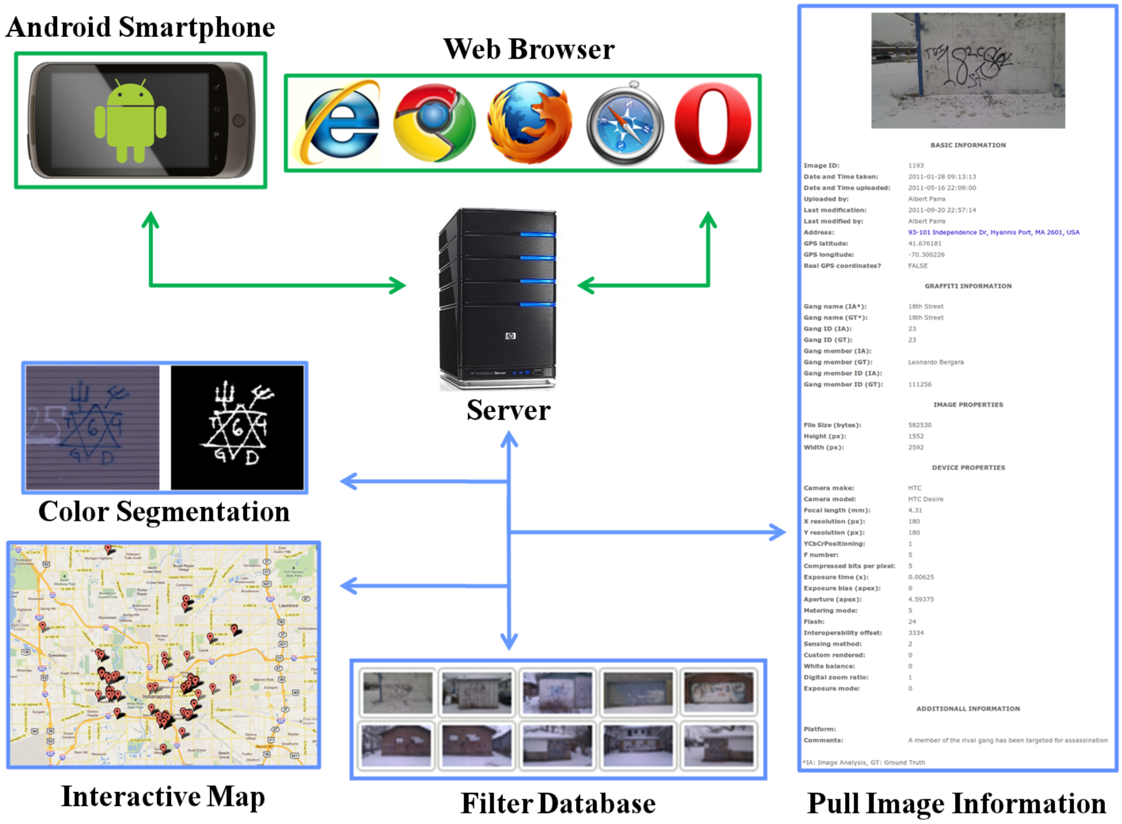


Fig. 3.1.: Overview of the GARI System - Client-Side Components (green) and Server-Side Components (blue).

3.2 Database of Gang Graffiti

In this section we describe how the image database is organized. We will first describe the database schema and then show by an example how the information GARI acquires is added to the database. The database of gang graffiti was deployed for three reasons:

1. To collect and organize graffiti images acquired by first responders. This includes the images, metadata, and any interpretation or other information provided by the first responder.

2. To store the results of the image analysis.

3. To manage first responders' credentials, allowing them to access the services available through the Android application and the web based interface.

Our database is implemented in PostgreSQL [73] on a Linux server. It consists of eight tables structured as shown in Figure 3.2. Note that the schema does not show all the fields in all the tables but just the relevant fields to indicate the association between the tables. Also the various IDs mentioned below (e.g. image ID) will be discussed in more detail after the tables are described in the following list.

1. **images**: Stores EXIF data from the images along with image location and general image information and the results from the image analysis. The fields related to this table are shown in Tables B.1, B.2, B.3 and B.4 in Appendix B.

2. **imageColors**: Stores all color IDs related to each image ID. This table is especially useful when more than one color is found in the same graffiti image.

3. **colors**: Stores the association between color IDs and color names.

4. **imageBlobs**: Stores the number of blobs in each graffiti, the ID of each component for each blob, and the color ID of each component. This also stores special attributes of components. These attributes may include a specific component being crossed-out, upside-down, etc. Table B.6 in Appendix B describes the fields of this table.

5. **blobComponents**: Stores the association between component IDs and component names, as well as the type ID for each component. Each component belongs to any of the following types: symbol, character, number, acronym, nickname, string.

6. **componentTypes**: Stores the association between type IDs and type names.

7. **gangComponents**: Stores the association between gang IDs and gang names, as well as the component ID (or multiple component IDs) associated with each

gang. This table is especially useful when more that one component is associated with the same gang name.

8. **users**: Stores users' credentials to access to the system services as well as information concerning administrative privileges, email addresses, and registration and login status. Table B.5 in Appendix B describes the fields of this table.

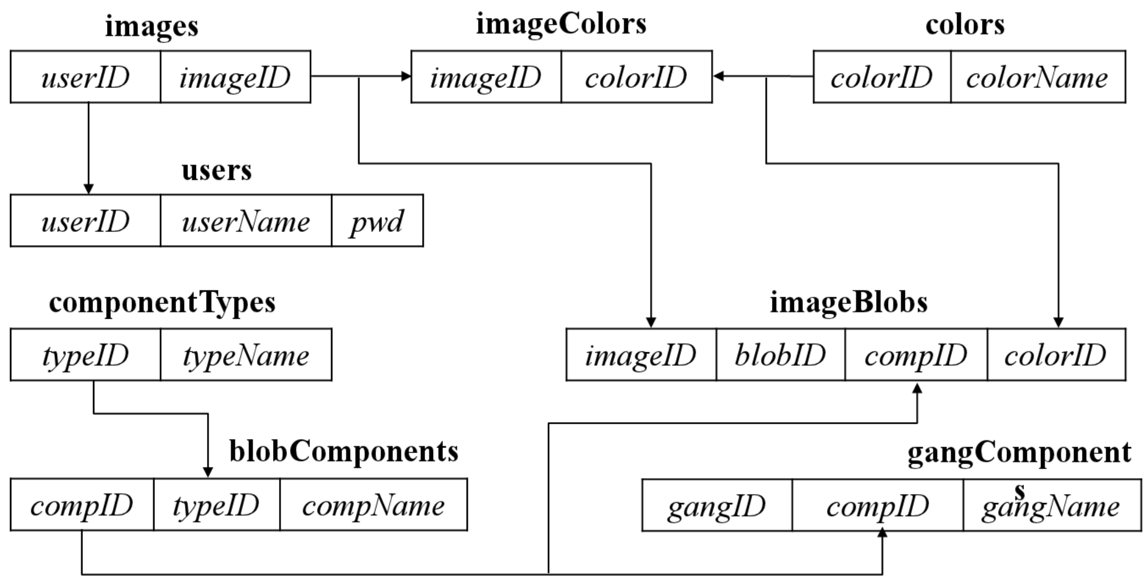


Fig. 3.2.: Database Schema Showing the Associations Between the Tables in the Database.

3.2.1 Adding Images to the Database

The following example illustrates the process of adding a graffiti image to the database. The image analysis is assumed to have been completed. Figure 3.3 shows the example image that has been manually labeled to facilitate the explanation. Each labeled circle represents a blob and each blob contains a distinguishable component of the graffiti. The blob labeling of the image corresponds with the field *blobID* from table *imageBlobs* in the database.

First, we fill table *imageColors* with the colors found in the graffiti. This is, black, green, and blue. Second, we analyze the blobs separately:

1. Color: black. Component: X3.

2. Color: green. Component: SPV.

3. Color: blue. Component: X3.

4. Color: blue. Component: LK. Crossed-out in green.

5. Color: blue. Component: ES. Crossed-out in green.

Note that the meaning of the acronyms and the type of the components is not addressed here. This information is assumed to already exist in the database.

Once the image analysis is complete the image, along with the blob information, is added to the database. Figure 3.4 shows the database fields filled with the information obtained from the graffiti in Figure 3.3. First, the user ID of the first responder who captured the image and the image ID are added to the *images* table. The image ID is a unique identifier of the graffiti image and it is automatically updated every time an image is uploaded to the server. Although it is not shown in Figure 3.4, some additional image information (i.e., EXIF data, GPS coordinates) is extracted from the uploaded image and added to the *images* table. Second, the color IDs for the three colors found in the graffiti, which are obtained by checking the color description field, (labeled *colorName* in Figure 3.4), are added to the *imageColors* table, and linked to the graffiti ID. At the same time, the five blobs are added to the *imageBlobs* table. Each blob has a corresponding component ID, which is obtained by checking the component description field, (labeled *compName* in Figure 3.4), of the *blobComponents* table. Each component has a color associated with it and can activate one or many attributes in the same table (see Table B.6 for all the attributes). In this example, blobs one to three do not have any additional attribute. Blobs four and five have activated the crossed-out attribute.

Note that this process is totally objective. That is, the information uploaded to the database does not require any interpretation from the first responder. With all the objective information available in the tables and the associations between the data one can produce an informed graffiti interpretation. For example, we have added components with IDs 27 (SPV) and 29 (LK). These IDs are associated with specific gang names in the *gangComponents* table. The same reasoning could be used if the graffiti did not contain any specific content with just the graffiti color being identified. Additional tables can relate gang IDs with color IDs effectively providing the results of gangs matching the specific color or colors.

Fig. 3.3.: Example of Graffiti (manually labeled).

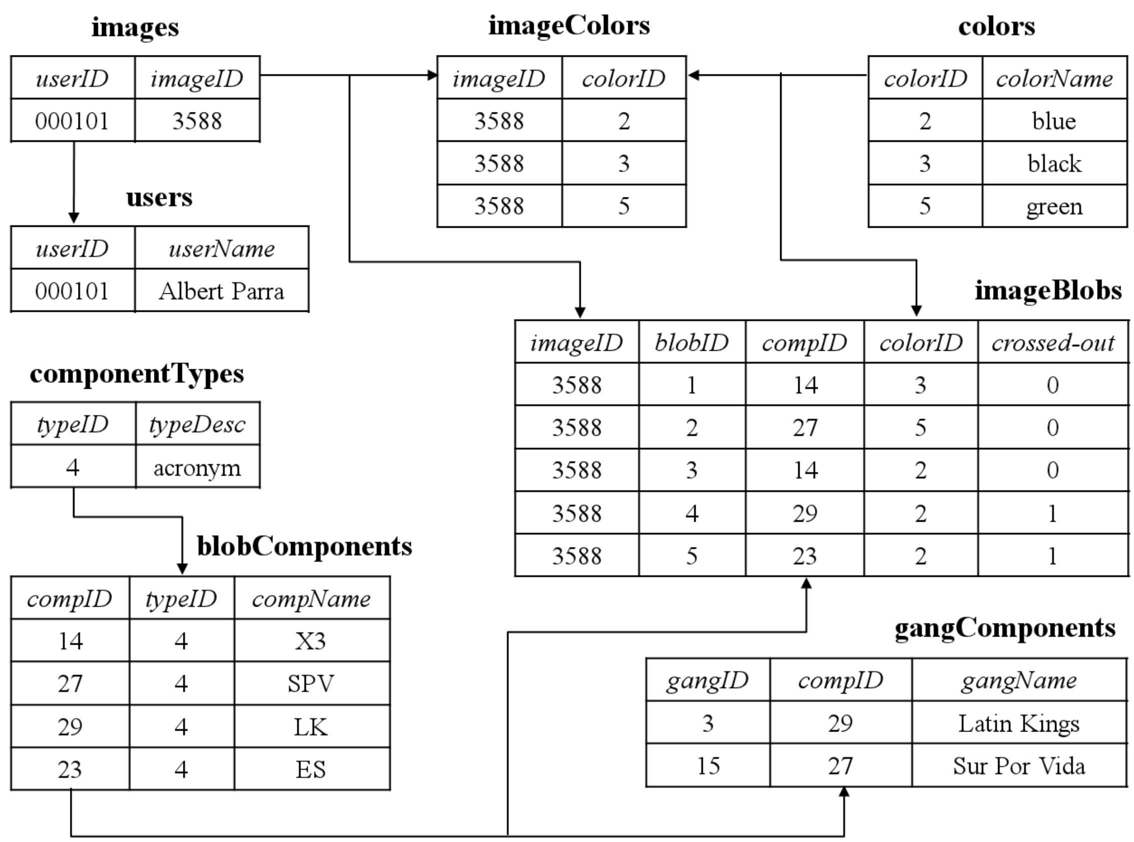


Fig. 3.4.: Database Fields with Information from the Graffiti in Figure 3.3.

3.3 Android Implementation

We implemented the GARI system on an Android device as summarized in Figure 3.5. In this section we describe how the Android application works and describe its user interface.

3.3.1 Overview

A user takes an image of the gang graffiti using the embedded camera on the device via the Graphical User Interface (GUI). The EXIF data of the image, including GPS location and date and time of capture, is automatically added to the image header. The user can then choose to upload the image to the server to be included in the database of gang graffiti or do color recognition. The first option, uploading to the server, allows the user to send the image and the EXIF data to the server creating a new entry in the database. The second option, color recognition, allows the user to trace a path in the current image using the device's touchscreen. The color in the path is then automatically detected (Section 2.2.2) and the result is shown to the user. The database of gang graffiti can then be queried to retrieve graffiti images of the same color.

Another option is to browse the database of gang graffiti given various parameters such as the distance from current location or date and time. The thumbnail images that match the query are downloaded from the server and shown to the user on the mobile telephone. The user can then browse the results to obtain more information about the specific graffiti. Note that in order to browse the database of gang graffiti a network connection is required.

We implemented the system on a HTC Desire mobile telephone running version 2.2 of the Android operating system. Tables 3.1 and 3.2 show the specifications of the Android OS and the HTC Desire, respectively.

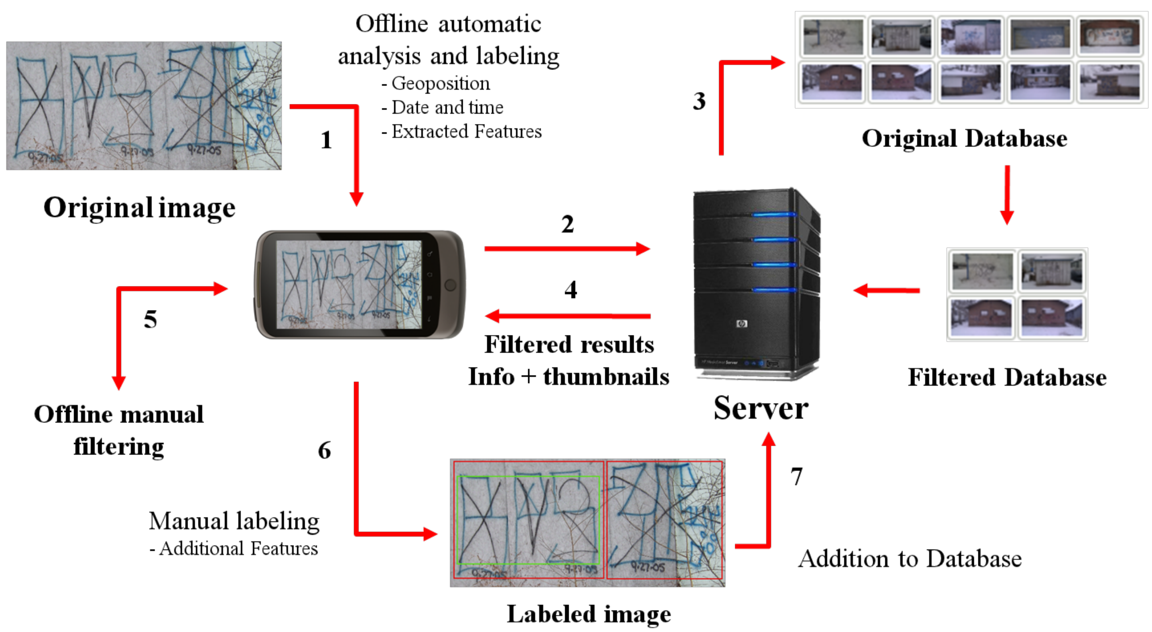


Fig. 3.5.: Overview of the GARI System.

Table 3.1: Main Features of the Android 2.2 Froyo OS.

Feature	Description
Kernel	Linux kernel 2.6.32, with support for RAM > 256MB SDIO scheduling
Performance and speed	Dalvik JIT compiler
Browser	V8 JavaScript engine
Camera API	∼20FPS Portrait and landscape orientation Camera settings (focal length, exposure, zoom level, view angle, and others)
Other	Application installation on external storage

3.3.2 User Interface

Our Android application does not require the use of a network connection. However it is mandatory if the user wants to browse the graffiti database or upload images to the graffiti database. A user must be assigned a User ID (equivalent to a First Responder ID) and a unique password in order to use GARI.

Table 3.2: Main Features of the HTC Desire Mobile Telephone.

Feature	**Description**
Operating system	Android 2.2 Froyo
CPU	1 GHz Qualcomm QSD8250 Snapdragon
GPU	Adreno 200 (AMD Z430)
Battery life	Up to 6 hours of talk time Up to 340 hours of standby time
Memory	576 MB RAM 512 MB flash storage (150 MB user accessible) 32 GB removable storage (microSDHC)
Data inputs	Dual-touch screen Ambient light sensors A-GPS Digital compass
Display	3.7-inch 480x800 WVGA AMOLED Capacitive touchscreen
Camera	5 Megapixels (2592 x 1944 pixels) Autofocus LED flash Geotagging
Connectivity	Wi-Fi 802.11b/g, 3G
Other	Flash 11.0 enabled

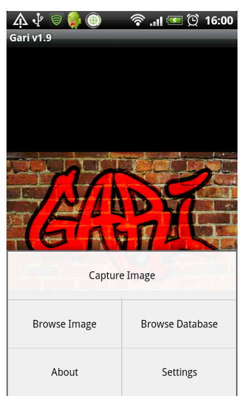
In this section we make the distinction between hardware and software keys in the application to describe some of the actions. Figure 3.6 illustrates this difference. Basically, hardware keys belong to the device, while software keys are created by the application.

Figure 3.7 illustrates the main screen of the Android application. The user can interact with the device and the database of gang graffiti using the following options. Some are only available when the user has captured or browsed an image.

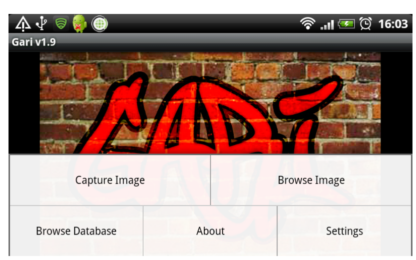
- Capture Image
- Browse Image
- Browse Database
- Settings

- Send to Server
- Trace Color
- Manual Input

Fig. 3.6.: Difference Between Hardware and Software Keys on the HTC Desire Mobile Telephone.



(a) Main Screen (portrait)



(b) Main Screen (landscape)

Fig. 3.7.: Main Screen of the GARI Application (most of the layouts support both orientations).

Capture Image

The menu option "Capture Image" starts the image capture. The user can then take an image of the graffiti as shown in Figure 3.8. Once the image is acquired, the user can choose either to retake the image or to tap "Done" to continue with the process.

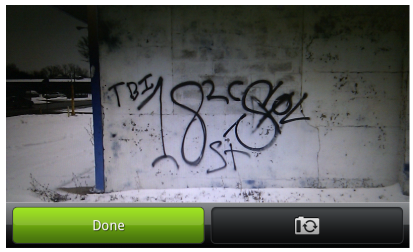


Fig. 3.8.: Image Capture.

The application then checks for the device location automatically in order to add the GPS coordinates to the image. Depending on the location system used (Network(GSM/WiFi) or GPS), it can take up to 30 seconds to acquire the location. The user is notified during the process as shown in Figure 3.9.

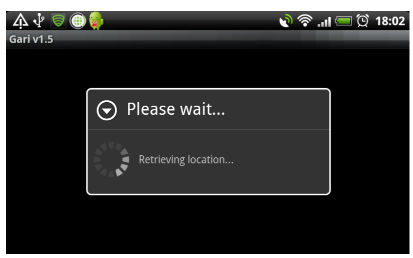
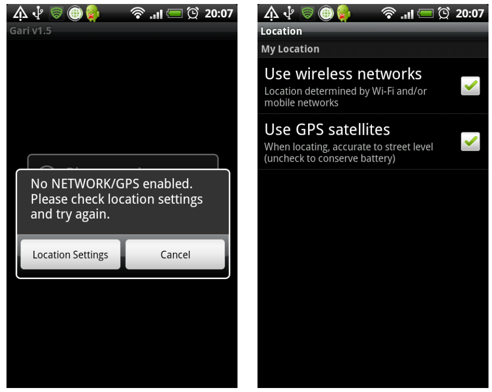


Fig. 3.9.: Progress Dialog - Location Retrieval.

If location services are not enabled on the device, the user is notified and taken to the location settings (Figure 3.10) where the location system can be enabled.

If, despite having the location systems enabled, the location system times out, a dialog will notify the user that the location cannot be determined. It also recommends the user to manually save the location information. Figure 3.11 shows the dialog.



(a) (b)

Fig. 3.10.: Dialog Indicating Locations Services are Not Enabled.

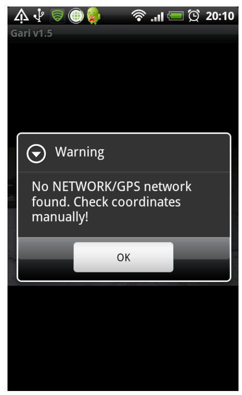
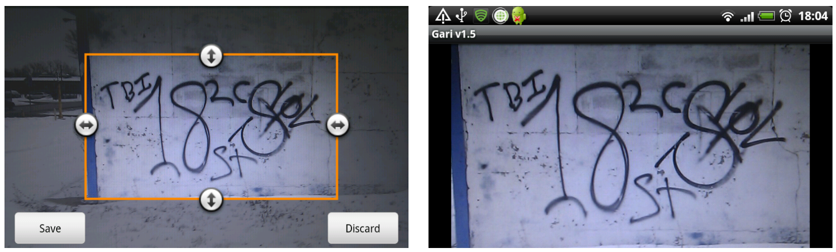


Fig. 3.11.: Dialog Box Notifying the User that No Location Can Be Found.

When the location has been acquired (automatically or manually), the user is given the option to crop the image (Figure 3.12). The image can be cropped by scaling the orange rectangle as shown in Figure 3.13a. When the desired graffiti contents are contained in the cropped area, the image can be cropped by tapping “Save.” The user will be returned to the main screen and the image will be set as background (Figure 3.13b).

Fig. 3.12.: Crop dialog.



(a) Crop Activity (b) Cropped image

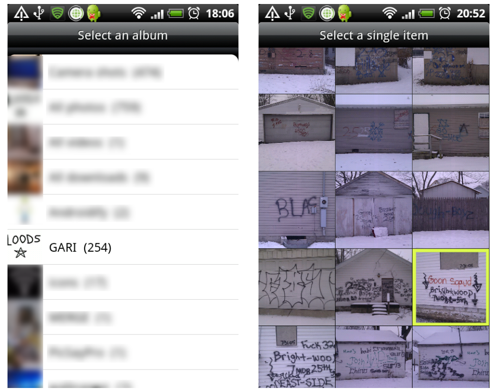
Fig. 3.13.: Image Cropping.

Browse Image

If an image has been acquired without using the GARI application, or if it has been taken with the the GARI application but not analyzed or sent to the server, the user has the option “Browse Image.” When tapped, a directory browsing window is opened and the user can search and select the desired image. Figure 3.19 shows an example of browsing.

Browse Database

The menu option “Browse Database” allows the user to browse the graffiti database by radius or distance from their current location. The system finds in the database the images in a chosen radius from the current location. Figure 3.15 shows the dialog where the user can select a radius between 1 mile and 20 miles. If the option “All” is selected the application use all the images in the database.



(a) (b)

Fig. 3.14.: Example of Image Browsing.

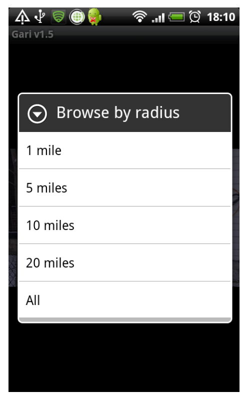
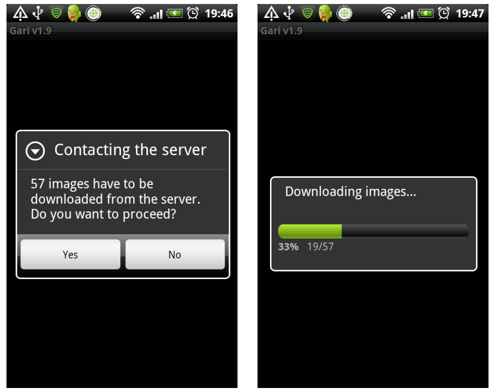


Fig. 3.15.: Browse by Radius.

If a specific radius is chosen, the application has to first acquire the user's current location. Then, the application contacts the graffiti server and checks how many graffiti thumbnail images have to be downloaded (Figure 3.16a). If the user accepts the information that matches the query is retrieved (Figure 3.16b). Figure 3.17 shows an example of the results, where each line contains a thumbnail image of a graffiti and basic information about it, including the date and time the image was acquired, and its GPS latitude and longitude.

To obtain more information about a particular graffiti, the user can tap either the thumbnail or the text field, and the application will contact the server, retrieving a larger image and the information available. Figure 3.18 shows an example of the



(a) (b)

Fig. 3.16.: Screen notifying the user that graffiti are to be downloaded from the server (3.16a) and that the thumbnail images are being retrieved (3.16b).

Fig. 3.17.: Results After Querying the Graffiti Database.

extended results. Currently, the text field includes the information shown in Tables B.1 to B.4 in Appendix B.

The menu hardware key has the option "Show in map". This allows the user to see the location of multiple (Figure 3.19a) or a single (Figure 3.19b) graffiti on a map.

Send to Server

Once the user captures an image using the "Capture Image" menu option or browses an image from the device using the "Browse Image" menu option, the menu

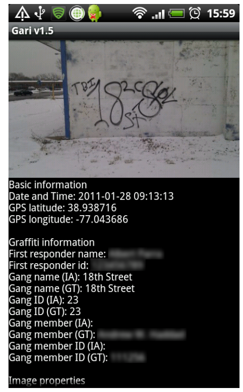
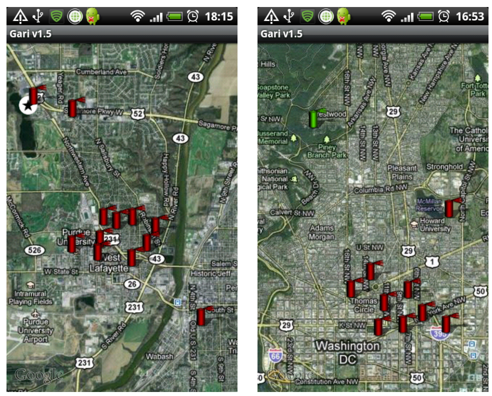


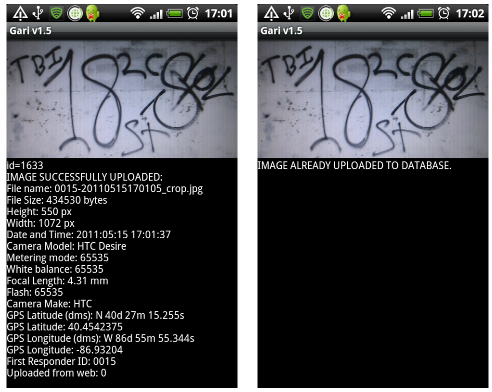
Fig. 3.18.: Extended Results After Querying the Graffiti Database.



(a) (b)

Fig. 3.19.: Graffiti Locations Displayed On A Map (black star in white circle is the current location).

option "Send to Server" is enabled in the main screen. This option allows the user to send the current image to the graffiti server. The application will contact the server and check if the image has been previously uploaded. If not, the image is sent to the server. Figure 3.20 shows both cases. Note that the image name includes the user ID and the date and time the image is uploaded. If the image is successfully added to the graffiti database the application will indicate this to the user. The information that can be displayed in the text field is shown in Tables B.1 to B.4.



(a) (b)

Fig. 3.20.: Result of Uploading an Image to the Server. The image can be upload successfully (3.20a), or already uploaded to database (3.20b).

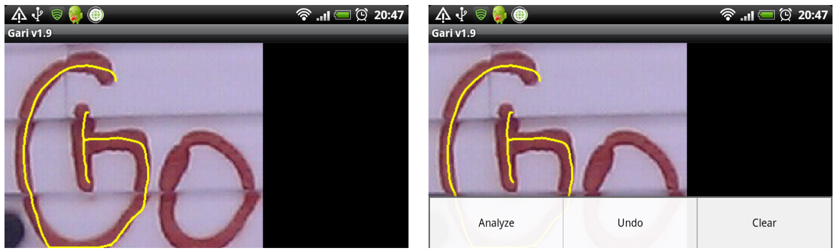
Trace Color

The menu option "Trace Color" allows the user to execute the application for detecting the graffiti color. This option is only enabled once an image has been captured or browsed. First the user has to select a region of the image containing the graffiti color as shown in Figure 3.21.

(a) (b)

Fig. 3.21.: Selecting the Region For Color Analysis.

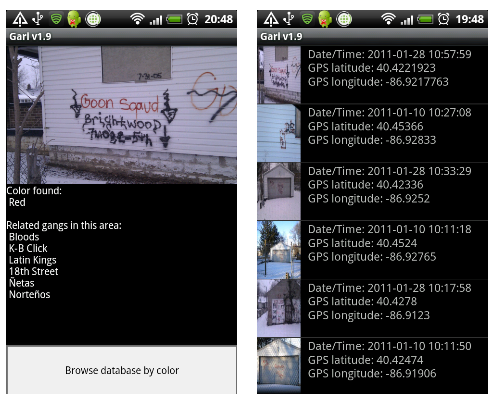
When the desired area is selected and "Save" is tapped, the user can create a path on the image using their finger as shown in Figure 3.22. There is no need to trace the entire area with the same color. A significant sample is enough to determine the color. Figure 3.22a shows an example. Figure 3.22b shows the available options color tracing. Since multiple paths can be defined, the "Undo" option removes the last path created. the "Clear" option clears all the paths created. The "Analyze" option extracts the current path and analyzes the color.



(a) Finger tracing. (b) Trace color options.

Fig. 3.22.: Tracing a Path On The Image.

Figure 3.23a shows the result of the color tracing. The application returns from the database the gangs that match the detected color. There is also the option "Browse database by color," which queries the database and finds the graffiti images that match the traced color. Figure 3.23b shows an example.



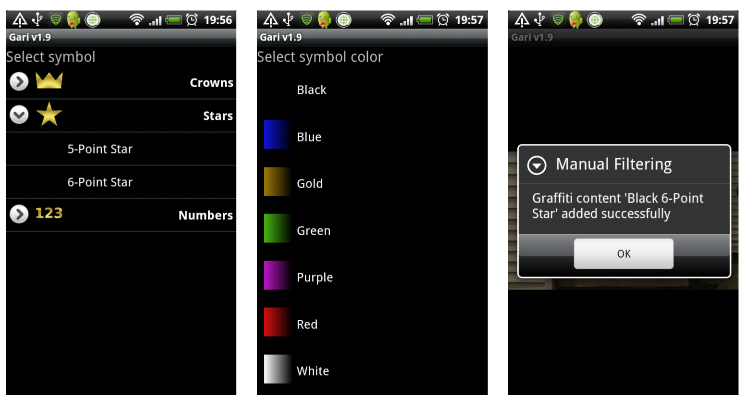
(a) (b)

Fig. 3.23.: Gangs Related to the Traced Color (3.23a) and the Graffiti Images in the Database that Match the Traced Color (3.23b).

Manual Input

The user can analyze the graffiti using the option “Manual Input.” This options allows a user to annotate the image manually and then store the information in the database. Figure 3.24 shows a graffiti image taken by a first responder. This menu option allows the user to select a symbol and its color. Figure 3.25 illustrates the manual input steps that identify the black 6-point star in Figure 3.24. This process can be repeated for any symbol, number, or color in the graffiti. The information is saved and later sent to the database.

Fig. 3.24.: Graffiti Image Containing a Black 6-Point Star.



(a) (b) (c)

Fig. 3.25.: Manual Annotation Steps For Identifying the Black 6-Point Star in Figure 3.24.

3.3.3 Security

Our Android application is used by first responders from multiple agencies. Therefore, it is mandatory to ensure that only authorized users can access and use the application. The connections to the server must be secure and all the information transmitted to and from the server must be encrypted (using the SSL/TLS protocol). The user credentials are sent every time the application contacts the server to make sure the connection is made by an authorized user. An additional level of security includes the creation of two types of users:

- **Regular users**: Can switch between users, change their password, delete specific images only taken by themselves, and send crashlogs.
- **Administrative users**: Can modify the server domain name/IP address, change user IDs, change passwords, delete specific images from any user, delete all images of any specific user, and send crashlogs.

When launching the GARI application a dialog box automatically prompts the user for login credentials (Figure 3.26). The user is required to input a user ID and a password.

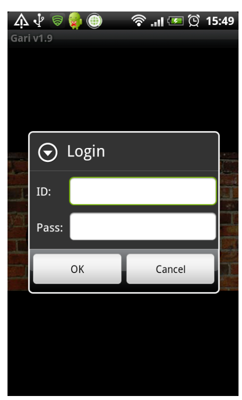


Fig. 3.26.: User ID Prompt.

The first time a user logs in the credentials are checked with the server and once they are validated they are stored in the device in an encrypted file. This allows the user to use the application without needing a network connection. Note that passwords are never stored as plaintext, neither on the device or the server. They are hashed using an MD5 cryptographic hash function [74]

All authorized users can access the “Settings” option from the main screen of the application. Figure 3.27 shows the various options. Note that no one can delete images from the server. At this time no one can edit the attributes of images retrieved from the server.

- **Server domain/IP**: Specifies the domain name or IP address of the server. The server is contacted to send images, browse the contents of the database, as well as to log in and change the login password. This option is only available for administrative users.
- **Switch user**: Allows a different user to log in and use the application.
- **Change password**: Allows the password of the current user to be changed. The new password is hashed and stored back in a system encrypted file for future logins. Note that this requires the application to contact the server requiring a network connection.
- **Delete specific images**: Allows specific images acquired by the user to be deleted. Administrative users can see and delete other users’ images.
- **Delete user’s images**: Allows all images taken by the user to be deleted. Administrative users can delete images from other users.
- **Delete all images**: Deletes all images. This option is only available for administrative users.
- **Send crashlog**: Allows the user to send crash feedback to the server when the application crashes. This helps us analyze and keep track of application errors.

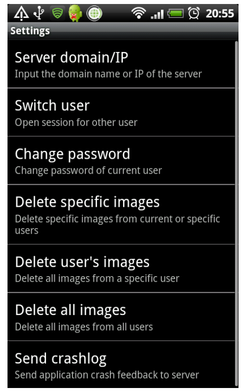


Fig. 3.27.: The Settings Screen.

3.4 Web Interface

3.4.1 System Overview

We also implemented our system as a web interface that gives a user access to the graffiti in the database and provides the ability to upload, modify and browse most database contents as summarized in Figure 3.28. The user logs in into the “Archive” using authorized credentials. Note that the credentials are the same for both the Android application and the web services. The user can then either browse the database of gang graffiti or upload an image. If the choice is to browse the database, the user can check the graffiti images and their attributes or filter the database using parameters such as radius from a specific location or address, capture data, upload data, or modified date. The results are shown as a list of thumbnail images with basic information that identifies the graffiti image. The user can then browse specific images and place them on a map, so to visually track gang activity. If the choice is to upload an image, the user can select a graffiti image from their local system (i.e., any device with a web browser). Some attributes can be adjusted through guided steps before adding the information to the database, such as location, gang information, or additional comments.

The web interface is available from any device with a web browser. This includes all desktop and laptop machines and all mobile telephones capable of browsing the

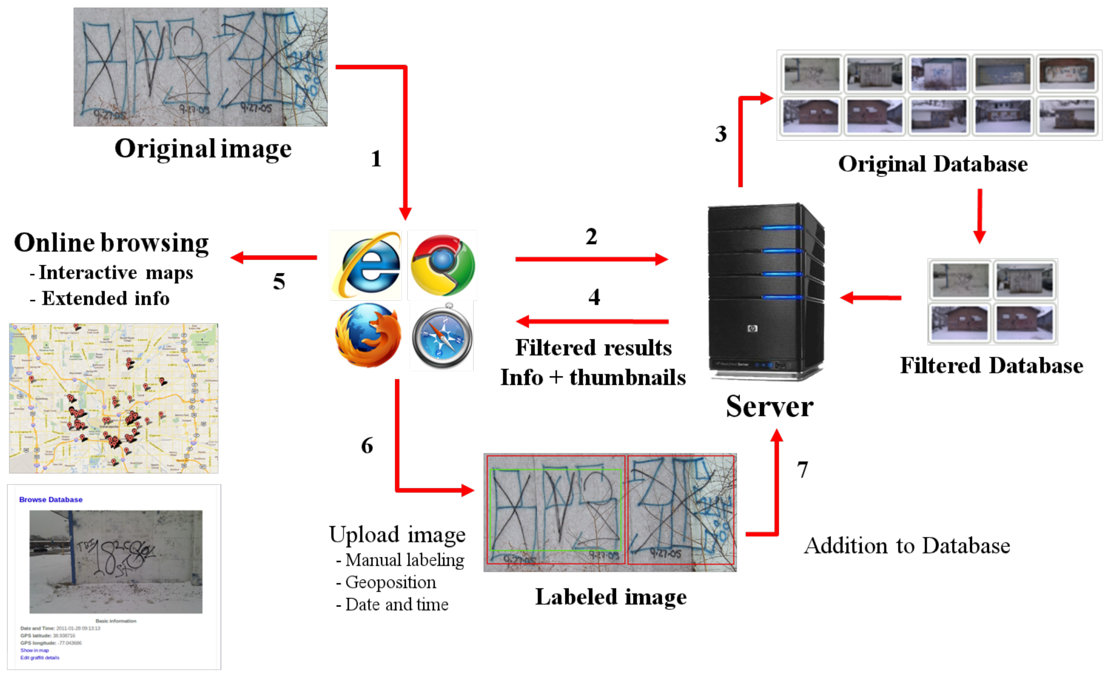


Fig. 3.28.: Overview of the Web Interface of the GARI System.

web (e.g., iPhone, Blackberry, Android devices). In some cases, the current location of the user is required in order to retrieve results from the database of gang graffiti such as when using the "radius" function to display graffiti on a map. Geolocation was introduced with HTML5 and it is widely implemented by many modern browsers. However, only the latest browsers support this service. Table 3.3 lists the browsers and their support level for Geolocation.

Table 3.3: Web Browsers Supporting HTML5 Geolocation Service.

Browser	Version
Firefox	3.5+
Internet Explorer	9+
Google Chrome	5+
Safari	5+
iPhone Safari	+3.0 OS
Android	Through Gears API
Opera	10.6+

3.4.2 User Interface

As of November 2011 the GARI website is located at https://redpill.ecn.purdue.edu/∼gari. The main page contains information about the GARI project, its principal investigators, and the graduate students involved. Figure 3.29 shows a snapshot.

Through the "Archive" link on the left sidebar the user can browse the entire database of gang graffiti. Figure 3.30 shows the available options once the user has logged in using authorized credentials. This include "Browse database" and "Upload image."

Browse database

The "Browse database" page (Figure 3.31) allows the user to either browse the entire graffiti database or to do a specific search given parameters. This include:

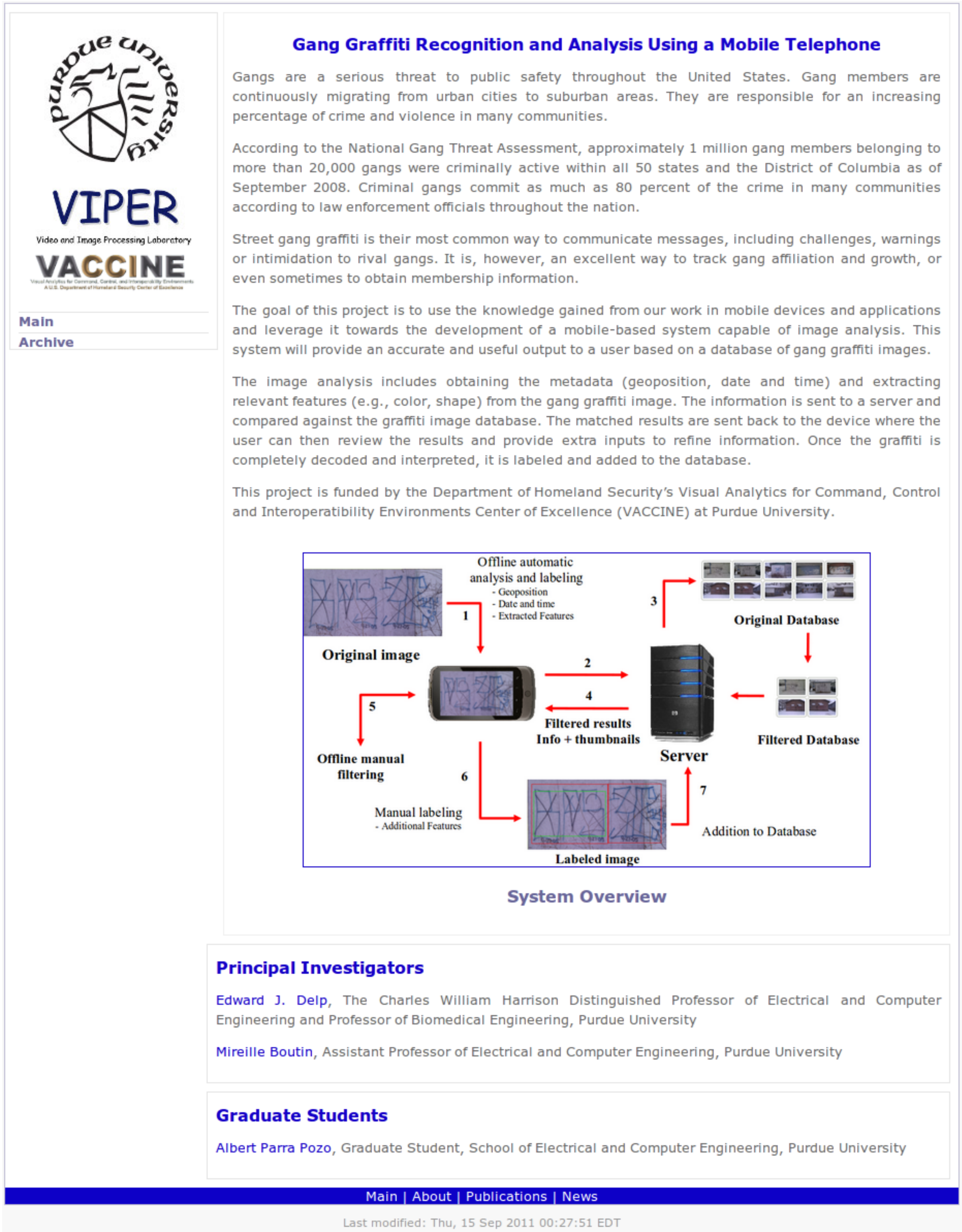
- **Search by radius**: Retrieves from the database all the graffiti in a specific radius from a location from the list. The locations in the list include the user's current location, the Video and Image Processing Laboratory (VIPER) at Purdue University, and the Indianapolis Metropolitan Police Department (IMPD). The "Current location" option requires the user to share their current location as shown in Figure 3.32.
- **Search by Date**: Retrieves the graffiti images captured, uploaded or modified in a specific period of time.
- **Search by address**: Retrieves the graffiti in a specific radius, from a specified address. Gives more flexibility than the "Search by radius" option.

The results are formatted as shown in Figure 3.33. At first, only a small-scale image and basic information is displayed. Depending on the search different parameters are shown including:

- Date/Time captured (uploaded, modified): date and time the image was taken, uploaded or modified, depending on the search.
- Address: address where the image was acquired. A map showing the graffiti location is opened when clicked.
- More information: link to show additional information concerning the graffiti.
- Image ID: image identifier in the database.
- Distance: distance from the user's current location to the graffiti. Only available when searching by radius or address.

Each graffiti or group of graffiti can be placed on an interactive map to visually track the results of a search. Figure 3.34 shows an example of the interactive map when a single graffiti is displayed. The graffiti is placed on a map, a balloon pops out, showing a thumbnail image and some information about the graffiti, including the date and time it was acquired and its location in GPS coordinates. Figures 3.35 and 3.36 show an example of the interactive map when multiple graffiti are displayed. Each marker represents the location of a graffiti from the search results. From this map the user can click on any of the markers to see a thumbnail of the graffiti image, its location in GPS coordinates, and a link to obtain more information about the graffiti. Figure 3.37 shows an example.

When "More information" is clicked, either from the list of results or from the interactive map, the user can see the information available in the database for the specific graffiti. Figure 3.38 shows an example. Currently, the extended information shown includes the fields in Tables B.1 to B.4.



Gang Graffiti Recognition and Analysis Using a Mobile Telephone

Gangs are a serious threat to public safety throughout the United States. Gang members are continuously migrating from urban cities to suburban areas. They are responsible for an increasing percentage of crime and violence in many communities.

According to the National Gang Threat Assessment, approximately 1 million gang members belonging to more than 20,000 gangs were criminally active within all 50 states and the District of Columbia as of September 2008. Criminal gangs commit as much as 80 percent of the crime in many communities according to law enforcement officials throughout the nation.

Street gang graffiti is their most common way to communicate messages, including challenges, warnings or intimidation to rival gangs. It is, however, an excellent way to track gang affiliation and growth, or even sometimes to obtain membership information.

The goal of this project is to use the knowledge gained from our work in mobile devices and applications and leverage it towards the development of a mobile-based system capable of image analysis. This system will provide an accurate and useful output to a user based on a database of gang graffiti images.

The image analysis includes obtaining the metadata (geoposition, date and time) and extracting relevant features (e.g., color, shape) from the gang graffiti image. The information is sent to a server and compared against the graffiti image database. The matched results are sent back to the device where the user can then review the results and provide extra inputs to refine information. Once the graffiti is completely decoded and interpreted, it is labeled and added to the database.

This project is funded by the Department of Homeland Security's Visual Analytics for Command, Control and Interoperatibility Environments Center of Excellence (VACCINE) at Purdue University.

System Overview

Principal Investigators

Edward J. Delp, The Charles William Harrison Distinguished Professor of Electrical and Computer Engineering and Professor of Biomedical Engineering, Purdue University

Mireille Boutin, Assistant Professor of Electrical and Computer Engineering, Purdue University

Graduate Students

Albert Parra Pozo, Graduate Student, School of Electrical and Computer Engineering, Purdue University

Main | About | Publications | News

Last modified: Thu, 15 Sep 2011 00:27:51 EDT

Fig. 3.29.: Main Page of the Web Interface of GARI.

Fig. 3.30.: “Archive” Section of the Web Interface of GARI.

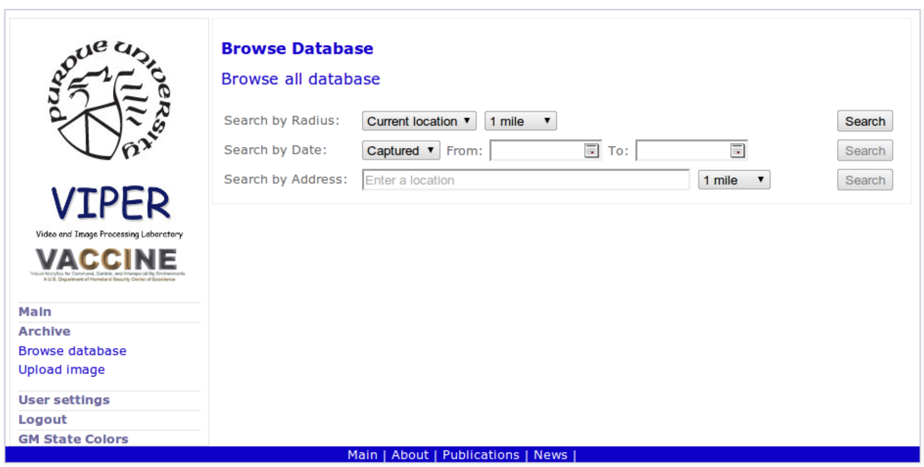


Fig. 3.31.: “Browse Database” Section of the Web Interface of GARI.

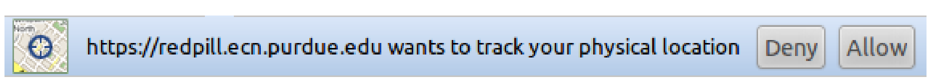


Fig. 3.32.: The Current Location of the User is only Acquired Upon Request.

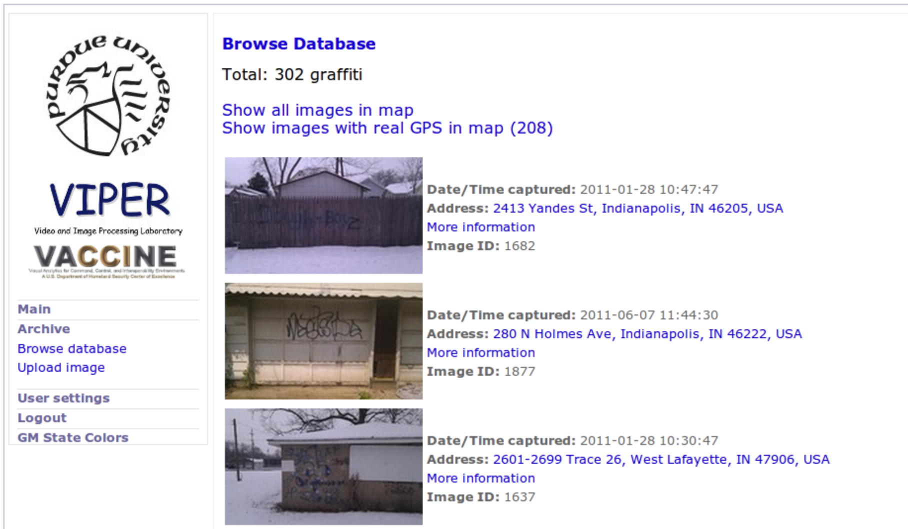


Fig. 3.33.: Results of Browsing The Database.

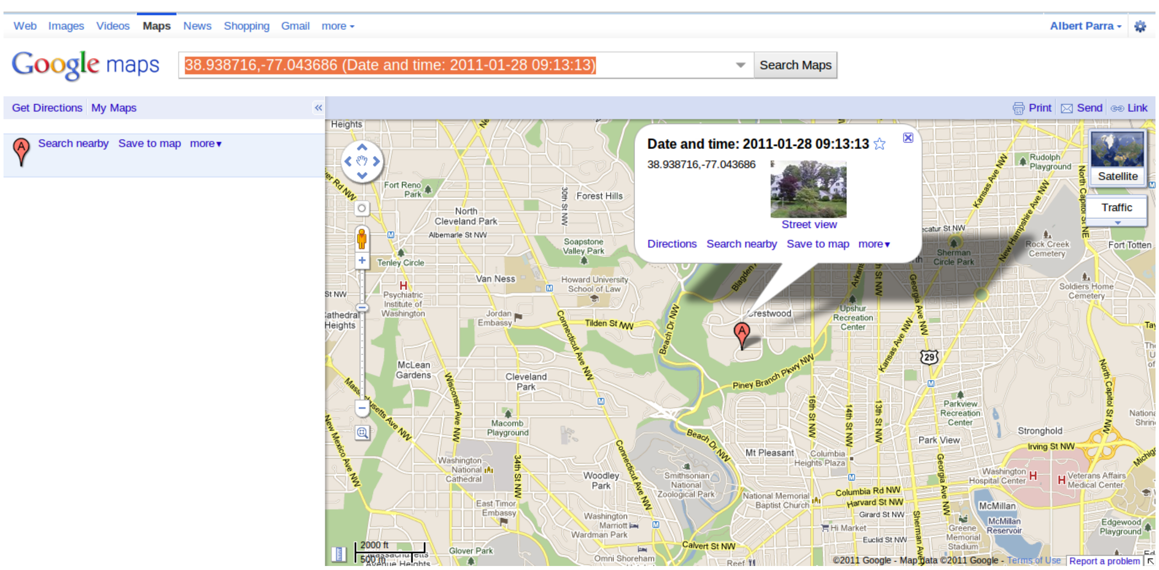


Fig. 3.34.: Example of the Interactive Map When a Single Graffiti is Displayed.

Fig. 3.35.: Example of the Interactive Map When Multiple Graffiti are Displayed.

Fig. 3.36.: If “Open in a new window” is Clicked, the Interactive Map Expands to Full Screen.

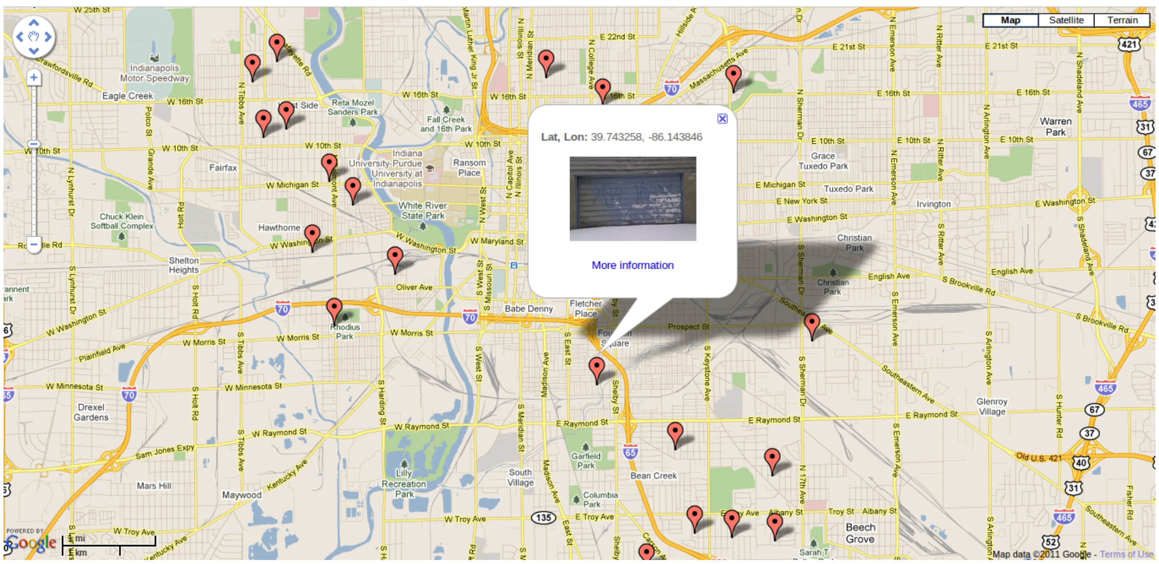
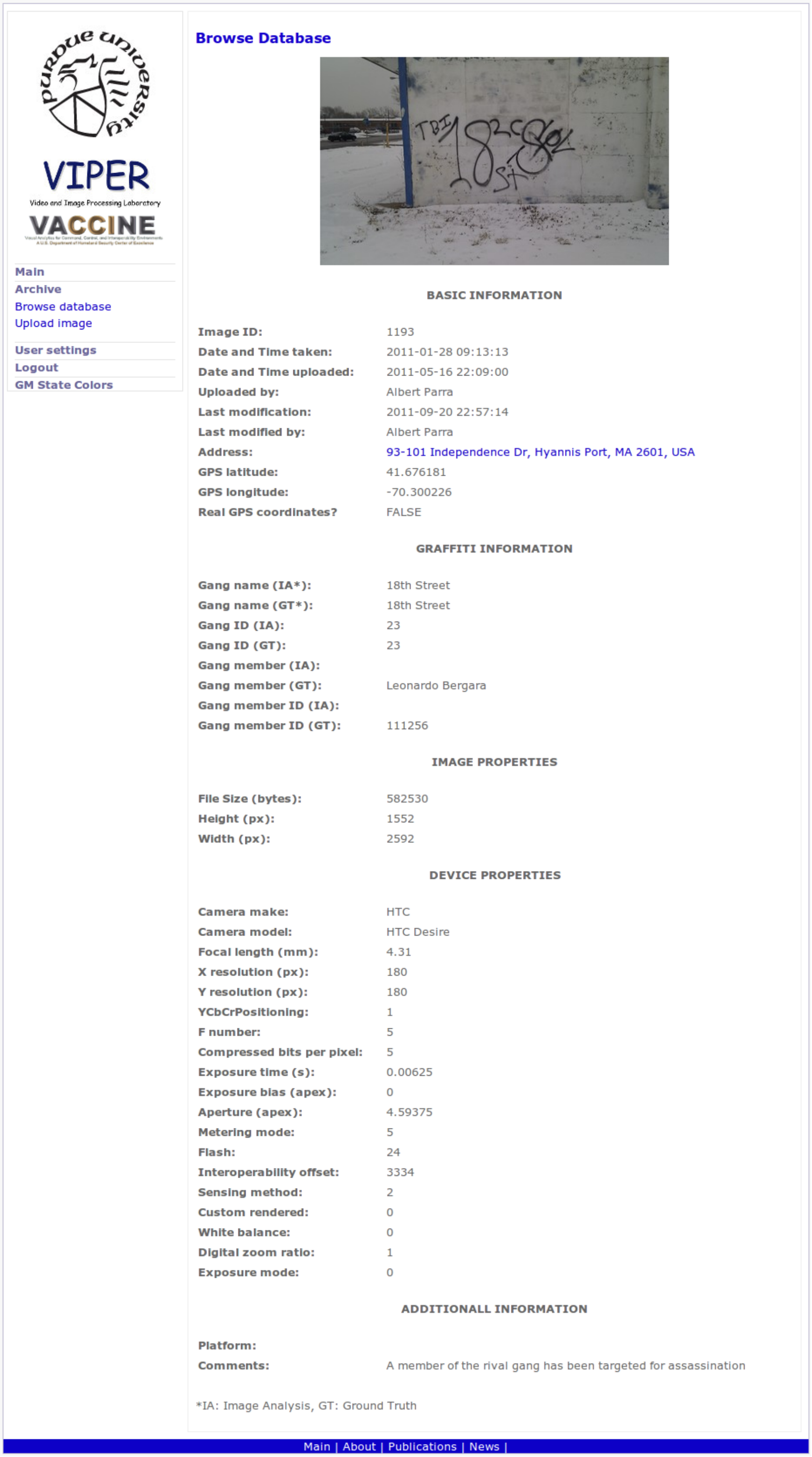


Fig. 3.37.: Example of a Popped Out Balloon in the Interactive Map when a Marker is Clicked.



Main
Archive
Browse database
Upload image
User settings
Logout
GM State Colors

Browse Database

BASIC INFORMATION

Image ID:	1193
Date and Time taken:	2011-01-28 09:13:13
Date and Time uploaded:	2011-05-16 22:09:00
Uploaded by:	Albert Parra
Last modification:	2011-09-20 22:57:14
Last modified by:	Albert Parra
Address:	93-101 Independence Dr, Hyannis Port, MA 2601, USA
GPS latitude:	41.676181
GPS longitude:	-70.300226
Real GPS coordinates?	FALSE

GRAFFITI INFORMATION

Gang name (IA*):	18th Street
Gang name (GT*):	18th Street
Gang ID (IA):	23
Gang ID (GT):	23
Gang member (IA):	
Gang member (GT):	Leonardo Bergara
Gang member ID (IA):	
Gang member ID (GT):	111256

IMAGE PROPERTIES

File Size (bytes):	582530
Height (px):	1552
Width (px):	2592

DEVICE PROPERTIES

Camera make:	HTC
Camera model:	HTC Desire
Focal length (mm):	4.31
X resolution (px):	180
Y resolution (px):	180
YCbCrPositioning:	1
F number:	5
Compressed bits per pixel:	5
Exposure time (s):	0.00625
Exposure bias (apex):	0
Aperture (apex):	4.59375
Metering mode:	5
Flash:	24
Interoperability offset:	3334
Sensing method:	2
Custom rendered:	0
White balance:	0
Digital zoom ratio:	1
Exposure mode:	0

ADDITIONALL INFORMATION

Platform:	
Comments:	A member of the rival gang has been targeted for assassination

*IA: Image Analysis, GT: Ground Truth

Main | About | Publications | News |

Fig. 3.38.: Extended Information for a Specific Search in the Database.

Upload Image

The "Upload image" page lets the user upload an image from the machine that is accessing the website. First, the user chooses the file to upload, which is previewed before actually adding any entries to the database of gang graffiti. Figure 3.39 shows an example.

Fig. 3.39.: Preview of an Image Before Uploading It to the Graffiti Database.

Once the image is uploaded, its EXIF data are automatically extracted, and a new entry is created in the database. If the device used to acquire the image did not have a GPS receiver, the location can be manually assigned in the next step shown in Figure 3.40. The user can input an address, which is geocoded and shown on an interactive map. More accurate GPS coordinates can be obtained by directly clicking on the interactive map. Besides the GPS coordinates other information is required in order to uniquely identify the image uploaded. This include the first responder name and first responder ID. Given that the user has accessed the database of gang graffiti using authorized credentials, the first responder ID is automatically obtained. The first responder name is associated to this ID in the *users* table in the database

(Table B.5 in Appendix B). Finally, the user can input additional information to help analyze the graffiti. This include fields for gang name, gang member, and additional comments. The gang name can be completed using a drop-down list of known-gangs, or entering it manually. The gang member is currently entered manually. The additional comments include any kind of information that does not fit in any of the previously described fields, such as symbols or colors found, graffiti meaning, or relative location of the graffiti with respect to the surrounding streets.

Clicking on "Submit Image" completes the editing step and shows the user the final output. This output is the same as when the user uses the "More Information" option when browsing the database.

3.4.3 Security

Access and navigation to the web interface are established and managed using encrypted Secure Sockets Layer (SSL) sessions. SSL encrypts information both during the transmission. The user must log in using authorized credentials before entering the archive. Figure 3.41 shows the login page. Once successfully logged in an SSL session is created and maintained for the current user. The user account can be managed by clicking on the "User Settings" link on the left sidebar. Note that currently the only option available is password change.

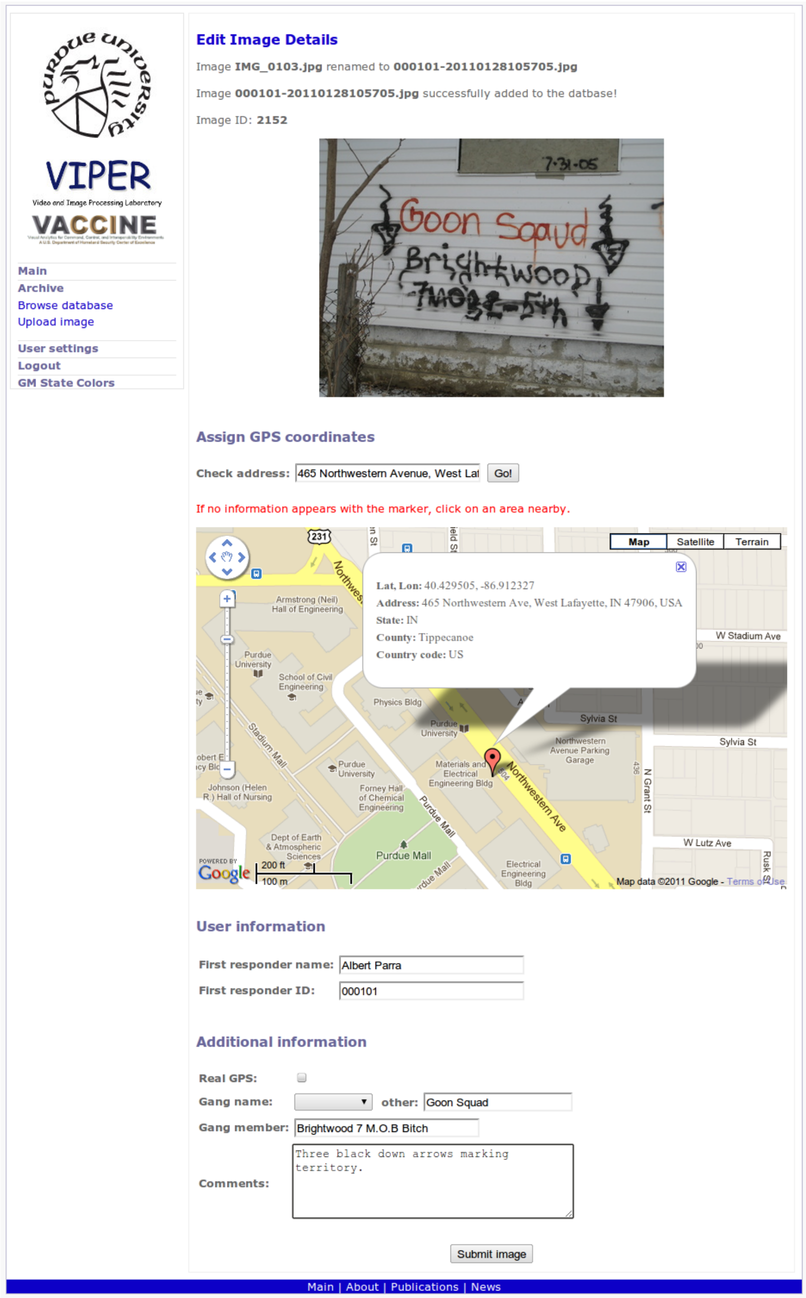


Fig. 3.40.: After Uploading the Image to the Database, information can be added.

Fig. 3.41.: Login Page for Accessing the Gang Graffiti Archive.

4. EXPERIMENTAL RESULTS

In this chapter, we investigate the execution time and memory size of our database of gang graffiti images as well as some of our processes on the hand-held device. We tested the execution time of color recognition based on the touchscreen tracing method on the HTC Desire Mobile Telephone with the Android 2.0 OS as described in the previous chapter. Since the other color analysis method we discussed in this thesis, color image segmentation based on Gaussian thresholding, runs on the server and does not perform complex-computing tasks, we did not consider running tests based on its execution time. Similarly, the battery life of the hand-held device is not yet considered an issue but may pose problems in the future if we add more processing functions to the Android application. Note that we have not performed any battery consumption tests, since we do not currently perform computationally intense operations on the HTC Desire.

4.1 Execution Time

We investigated the the execution time of several processes both on the hand-held device and on the server. First, we tested the transformation between color spaces using various approaches on the hand-held device. Second, we tested the query performance of our database on the server, by computing the time it takes to upload an image to the database of gang graffiti and receiving the results back to the hand-held device, after doing all the necessary operations on the server.

4.1.1 RGB to Y'CH Conversion

In Section 2.2.2 we described two approaches to transform the RGB color space to our Y'CH color space. The first, which we called *arithmetic approach* in Chapter 2, converts RGB to Y'CH by only doing arithmetic operations, while the second, which we called *trigonometric approach* in Chapter 2, converts to YIQ color space as an intermediate step, doing arithmetic and trigonometric operations. As a reminder, Equation 4.1 shows the mathematical definition of the arithmetic approach and Equation 4.2 shows the mathematical definition of the trigonometric approach. Note that Equation 4.2 does not define the transformation RGB to YIQ, since it is a linear transformation, it will not have an influence on the execution time of the overall transformation RGB to Y'CH.

$$\begin{aligned}
Y &= 0.299R + 0.587G + 0.114B \\
C &= max(R,G,B) - min(R,G,B) \\
&= M - n \\
H &= \begin{cases} 60(\frac{G-B}{C}) & \text{if M=R} \\ 60(\frac{B-R}{C} + 2) & \text{if M=G} \\ 60(\frac{R-G}{C} + 4) & \text{if M=B} \\ \text{undefined} & \text{if C=0} \end{cases}
\end{aligned} \tag{4.1}$$

$$Y = 0.299R + 0.587G + 0.114B$$
$$C = \sqrt{I^2 + Q^2}$$
$$H = \begin{cases} \arctan(\frac{Q}{I}) & I > 0 \\ \pi + \arctan(\frac{Q}{I}) & Q \geq 0,\ I < 0 \\ -\pi + \arctan(\frac{Q}{I}) & Q < 0,\ I < 0 \\ \frac{\pi}{2} & Q > 0,\ I = 0 \\ -\frac{\pi}{2} & Q < 0,\ I = 0 \\ \text{undefined} & Q = 0,\ I = 0 \end{cases} \tag{4.2}$$

Given that trigonometric operations are computationally more complex than arithmetic operations [75], we could assume that the arithmetic approach is always computationally faster than the trigonometric approach. However, we conducted tests to verify this. Table 4.1 and Figure 4.1 show the results of both transformations using various number of data points on the HTC Desire. Note that each data point corresponds to a pixel operation. Also note that the functions used to compute the time differential both on the hand-held device are accurate to the nearest millisecond. One can see how the execution time of the trigonometric approach grows exponentially faster than the arithmetic approach when the number of data points is greater than approximately one million. For example, for a five megapixel image (i.e., five million data points) the difference between the arithmetic approach and the trigonometric approach can be linearly interpolated to 3.36 seconds. Since the RGB to Y'CH conversion is done not only along a traced path during the color recognition process, but also on entire images during the image segmentation process, it is worth considering the arithmetic approach as a lightweight and fast approach if we plan on doing color image segmentation on the device in the future.

Table 4.1: Execution Time of the Arithmetic and the Trigonometric Approaches For Color Conversion.

	Execution Time	
Data Points	**Arithmetic**	**Trigonometric**
100	0 s	0 s
1,000	0.002 s	0.004 s
10,000	0.010 s	0.010 s
100,000	0.02 s	0.10 s
1 million	0.20 s	0.96 s
10 million	1.91 s	9.39 s
100 million	18.37 s	91.85 s
1 billion	183 s	922 s

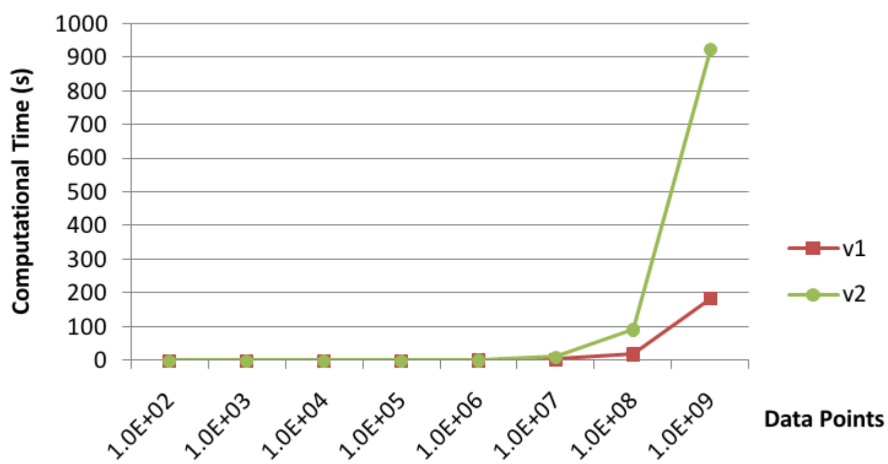


Fig. 4.1.: Execution Time with Respect to the Number of Data Points for the Arithmetic and the Trigonometric Approaches For Color Conversion.

4.1.2 Database Query Performance

We tested the elapsed time between sending an image from the hand-held device, using the Android application, and receiving the results of the upload. On the client side, the process includes sending and receiving the image to the server via HTTPS and returning the graffiti image thumbnail and text retrieved to the user. On the server side, the process includes creating a session for the user, checking image existence in the database, copying the image to a specific directory, creating the thumbnail image and reduced size copies of the image, extracting up to 24 EXIF data points from the image, creating a new entry in the PosgtreSQL table and adding information in as many as 30 fields, and sending back a string with the results of the upload. Table 4.2 shows the details of ten graffiti image uploads using the same network conditions (WiFi). As one can see most of the elapsed time is due to the HTTPS connection since the user interface operations on the hand-held device (for the specific action of uploading an image to the server) do not slow down the process.

Table 4.2: Elapsed Time On the Hand-Held Device and the Server When Uploading an Image.

Image Size	**Server Time**	**Total Time**
146.7 KB	0.66 s	2.24 s
157.9 KB	0.65 s	2.33 s
179.8 KB	0.65 s	2.66 s
203.3 KB	0.66 s	2.42 s
207.9 KB	0.64 s	2.44 s
227.8 KB	0.65 s	2.34 s
609.9 KB	1.05 s	3.64 s
639.8 KB	1.47 s	4.71 s
653.6 KB	1.06 s	4.00 s
760.4 KB	1.07 s	4.31 s

4.2 Memory Size

We computed the memory size of the images in the gang graffiti database as well as the memory size of the Android application.

4.2.1 Database of Gang Graffiti

We cooperated with the Indianapolis Metropolitan Police Department to acquire graffiti images. This allows us to be able to accurately calibrate and analyze the images. Currently we have 657 images from the city of Indianapolis. These include images acquired with and without using a tripod and with and without fiducial markers (see Section 2.1). We used three digital cameras for this purpose: a 10Mpx Canon Powershot S95, a 4Mpx Panasonic Lumix DMC-FZ4, and a 5Mpx HTC Desire (Android mobile telephone). From these images, 151 images are currently contained in the database of gang graffiti. The rest are for research purposes. As of November 2011, a total of 38 users have entered 153 graffiti images to our system, resulting in a total of 304 graffiti images that can be browsed from both the Android application and the web interface. The 304 images are stored in the server and all have a thumbnail and a reduced size version. This makes a total of 912 images adding up to 358 MB of data.

Taking into account the 506 images not in the database (containing fiducial marks - for research purposes) we have a total of 1,418 images adding up to 613 MB of data.

4.2.2 Android Application

The Android application on the hand-held device consists of an Android application package file (APK) of 1.8 MB. This includes the compiled code and the multimedia used when the user is not connected to a network. When contacting to the server the contents of the database can be browsed. If the users chooses to retrieve thumbnail images of all the 304 browsable images in the database, 2.75 MB are added to the application, making a total of 4.55 MB.

5. CONCLUSIONS AND FUTURE WORK

5.1 Contributions

In this thesis an integrated mobile system for gang graffiti image acquisition and recognition is described. Our methods include color recognition based on touchscreen tracing and color image segmentation based on Gaussian thresholding. We have also investigated the design and deployment of an integrated image-based database system. The main contributions in the area of image analysis are as follows:

- We described a color recognition method based on touchscreen tracing. This method includes a color space transformation from RGB to Y'CH which we believe is more intuitive and perceptually relevant.
- We presented a color image segmentation method based on Gaussian thresholding.

The main contributions in the design and deployment of the integrated image-based database system are as follows:

- We developed an integrated image-based database system where data from users and images are connected to gang graffiti information for analysis and tracking.
- We created a web-based interface for first responders and researchers, to upload images and browse gang related information by location, date and time, using interactive maps for better visualization. It is accessible from any device capable of connecting to the Internet, including iPhone and Blackberry.
- We created an Android application for first responders in the field, to upload images to the server, browse gang related information by location using interactive maps, and perform color recognition.

5.2 Project Status

As of November 2011 we have developed an Android application and a web-based interface for the GARI system. The Android application includes color analysis, images acquisition and upload, manual graffiti filtering, and database browsing through lists and interactive maps. The web-based interface includes the same capabilities except for the color analysis, and can be accessed from any device capable of connecting to the Internet (e.g., iPhone, Blackberry, laptop/desktop). Our current color analysis system includes two methods. First, color recognition based on touchscreen tracing, which is performed on the Android device. Second, color image segmentation based on Gaussian thresholding, which is performed on the server.

In our tests, the computational time for the RGB-Y'CH color space transformation for the color recognition on a hand-held device yields exponentially lower computational time using the arithmetic approach than using the cylindrical coordinates transformation approach. This result is significant when the number of data points to be analyzed is larger than the number of pixels in a 1Mpx image. Our tests on database query performance suggest that the bottleneck for the upload and retrieval process is the network connection. This is due to the fact that an entire image is sent to the server using HTTPS which highly depends on the network speed.

Our database of gang graffiti images contains 304 browsable images with associated thumbnails and reduced size versions. These 912 images are 358 MB of data. We have also acquired a total of 657 images for research purposes. The Android application has a memory size of 1.8 MB on the hand-held device. If all 304 thumbnail images from the database are downloaded for browsing, the application would take 4.55 MB of data.

5.3 Future Work

Our long term goal is to develop a system based on a mobile device such as a mobile telephone, capable of using location-based services, combined with image analysis, to automatically populate a database of graffiti images with information that can be used by law enforcement to identify, track, and mitigate gang activity. This can be done by implementing image analysis methods to segment the graffiti image in order to detect shapes, such as symbols and numbers, and orientation with respect to each other. These results can be associated to identify gangs, gang members, and track gang activity.

Our short term goal is to use our current color image segmentation method to create a probability map of the image to be used for shape detection. Our color recognition could be improved by building a more specific Y'CH color space, that is, to give specific angular weight to colors related to gangs and gang members.

Other future work includes enlarging the number of fields and relationships in the database so as to link gangs to their respective colors, acronyms, gang members, locations, or activity over time. The same can be done with graffiti components, in order to automatically interpret their position and alignment and the relationship between different components in the same or other graffiti. We also need to investigate the use of color calibration information for any type of future color analysis we do.

LIST OF REFERENCES

LIST OF REFERENCES

[1] National Drug Intelligence Center (NDIC), *Attorney General's Report to Congress on the Growth of Violent Street Gangs in Suburban Areas.* United States Deptartment of Justice, April 2008.

[2] Japan Electronic Industry Development Association (JEIDA), "Design Rule for Camera File System, version 1.0." 1998.

[3] D. Ley and R. Cybriwsky, "Urban Graffiti as Territorial Markers," *Annals of the Association of American Geographers*, vol. 64, no. 4, pp. 491–505, December 1974.

[4] J. Ferrell, *Crimes of Style: Urban Graffiti and the Politics of Criminality.* Garland, New York, 1993.

[5] W. Miller, *Crime by Youth Gangs and Groups in the United States.* U.S. Dept. of Justice, Office of Justice Programs, Office of Juvenile Justice and Delinquency Prevention, 1992.

[6] M. C. Morrone and D. C. Burr, "Feature Detection in Human Vision: A Phase-Dependent Energy Model," *Proceedings of the Royal Society of London. Series B, Biological Sciences*, vol. 235, no. 1280, pp. 221–245, December 1988, London, United Kingdom.

[7] G. Knox, C. Robinson, National Crime Gang Research Center (U.S.) Staff, and National Gang Crime Research Center (U.S.), *Gang Profiles: An Anthology.* New Chicago School Press, Incorporated, 2004.

[8] A. K. Jain, J.-E. Lee, and R. Jin, "Graffiti-ID: Matching and Retrieval of Graffiti Images," *Proceedings of the 1st ACM Workshop on Multimedia in Forensics*, pp. 1–6, October 2009, Beijing, China.

[9] W. Tong, J.-E. Lee, R. Jin, and A. K. Jain, "Gang and Moniker Identification by Graffiti Matching," *Proceedings of the 3rd ACM Workshop on Multimedia in Forensics and Intelligence*, November 2011 (to appear), Scottsdale, AZ.

[10] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, pp. 91–110, November 2004, Hingham, MA.

[11] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, "The FERET Evaluation Methodology for Face-Recognition Algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 1090–1104, October 2000, Los Alamitos, CA.

[12] "Graffiti Tracker." [Online]. Available: http://graffititracker.net

[13] "Military Specification (MIL)-STD-150A, Photographic Lenses," May 1959.

[14] E. A. Johnson, "Touch Display - A Novel Input/Output Device for Computers," *Electronics Letters*, vol. 1, no. 8, p. 219, 1965.

[15] E. A. Johnson, "Touch Displays: A Programmed Man-Machine Interface," *Ergonomics*, vol. 10, no. 2, pp. 271–277, 1967.

[16] N. Matsushita and J. Rekimoto, "HoloWall: Designing a Finger, Hand, Body, and Object Sensitive Wall," *Proceedings of the 10th Annual ACM Symposium on User interface Software and Technology*, pp. 209–210, 1997, Banff, Alberta, Canada.

[17] S. Izadi, H. Brignull, T. Rodden, Y. Rogers, and M. Underwood, "Dynamo: A Public Interactive Surface Supporting the Cooperative Sharing and Exchange of Media," *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology*, pp. 159–168, 2003, Vancouver, Canada.

[18] J. Rekimoto, "SmartSkin: An Infrastructure for Freehand Manipulation on Interactive Surfaces," *Proceedings of the 20th Annual SIGCHI Conference on Human Factors in Computing Systems*, pp. 113–120, 2002, Minneapolis, MN.

[19] A. Pirhonen, S. Brewster, and C. Holguin, "Gestural and Audio Metaphors As a Means of Control for Mobile Devices," *Proceedings of the 20th Annual SIGCHI Conference on Human Factors in Computing Systems*, pp. 291–298, 2002, Minneapolis, MN.

[20] E. Hoggan, S. A. Brewster, and J. Johnston, "Investigating the Effectiveness of Tactile Feedback for Mobile Touchscreens," *Proceedings of the 26th Annual SIGCHI Conference on Human Factors in Computing Systems*, pp. 1573–1582, 2008, Florence, Italy.

[21] K. S. Deoras, M. R. Wolfson, R. L. Searls, S. R. Hilfer, J. B. Sheffield, and T. H. Shaffer, "Use of a Touch Sensitive Screen and Computer Assisted Image Analysis for Quantitation of Developmental Changes in Pulmonary Structure," *Pediatr Pulmonol*, vol. 9, no. 2, pp. 109–18, 1990.

[22] G. Wyszecki and W. Stiles, *Color Science: Concepts and Methods, Quantitative Data, and Formulae*, ser. Wiley classics library. John Wiley & Sons, 2000.

[23] J. Krauskopf and G. Karl, "Color Discrimination and Adaptation," *Vision Research*, vol. 32, no. 11, pp. 2165–2175, 1992.

[24] K.-M. Cho, J.-H. Jang, and K.-S. Hong, "Adaptive Skin-Color Filter," *Pattern Recognition*, vol. 34, no. 5, pp. 1067–1073, 2001.

[25] R. González and R. Woods, *Digital Image Processing*, 2nd ed. New Jersey: Prentice Hall, 2002.

[26] R. Jusoh, N. Hamzah, M. Marhaban, and N. Alias, "Skin Detection Based on Thresholding in RGB and Hue Component," *Proceedings of the 2010 IEEE Symposium on Industrial Electronics Applications*, pp. 515–517, October 2010, Penang, Malaysia.

[27] C. Poynton, *Digital Video and HDTV Algorithms and Interfaces*, 1st ed. San Francisco, CA: Morgan Kaufmann Publishers Inc., 2003.

[28] M. Asmare, V. Asirvadam, and L. Iznita, "Color Space Selection for Color Image Enhancement Applications," *Proceeding of the International Conference on Signal Acquisition and Processing*, pp. 208–212, April 2009.

[29] M. Tkalcic and J. Tasic, "Colour Spaces: Perceptual, Historical and Applicational Background," *Proceedings of the IEEE Region 8 Eurocon 2003: Computer as a Tool*, vol. 1, pp. 304–308, September 2003, Ljubljana, Slovenia.

[30] G. H. Joblove and D. Greenberg, "Color Spaces for Computer Graphics," *Proceedings of the 5th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 20–25, August 1978.

[31] A. R. Smith, "Color Gamut Transform Pairs," *Proceedings of the 5th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 12–19, 1978, New York, NY.

[32] A. Hanbury, "A 3D-Polar Coordinate Colour Representation Well Adapted to Image Analysis," *Proceedings of the 13th Scandinavian Conference on Image Analysis*, pp. 804–811, June-July 2003, Halmstad, Sweden.

[33] J. D. Foley and A. Van Dam, *Fundamentals of Interactive Computer Graphics*. Boston, MA: Addison-Wesley Longman Publishing Co., Inc., 1982.

[34] M. Agoston, *Computer Graphics and Geometric Modeling: Implementation and Algorithms*, ser. Computer Graphics and Geometric Modeling. Springer, 2005.

[35] *Recommendation ITU-R BT.601, Encoding Parameters of Digital Television for Studios*, International Telecommunications Union, Geneva, Switzerland, 1992.

[36] J. D. Foley, A. van Dam, S. Feiner, and J. Hughes, *Computer Graphics: Principles and Practice*, 2nd ed. Reading, MA: Addison-Wesley, 1990.

[37] R. Paige, J. Reif, and R. Wachter, *Parallel Algorithm Derivation and Program Transformation*, ser. Kluwer International Series in Engineering and Computer Science. Kluwer Academic, 1993.

[38] R. M. Haralick and L. G. Shapiro, "Image Segmentation Techniques," *Computer Vision, Graphics, and Image Processing*, vol. 29, no. 1, pp. 100–132, 1985.

[39] K. Fu and J. Mui, "A Survey on Image Segmentation," *Pattern Recognition*, vol. 13, no. 1, pp. 3–16, 1981.

[40] Rosenfeld, Azriel and Kak, Avinash C., *Digital Picture Processing*, 2nd ed. Orlando, FL: Academic Press, Inc., 1982.

[41] N. R. Pal and S. K. Pal, "A Review on Image Segmentation Techniques," *Pattern Recognition*, vol. 26, no. 9, pp. 1277–1294, 1993.

[42] T. Q. Chen, Y. L. Murphey, R. Karlsen, and G. Gerhart, "Color Image Segmentation in Color and Spatial Domain," *Proceedings of the 16th International Conference on Developments in Applied Artificial Intelligence*, pp. 72–82, 2003, Laughborough, United Kingdom.

[43] W. Skarbek and A. Koschan, "Colour Image Segmentation - A Survey," Technical University of Berlin, Department of Computer Science, Tech. Rep., 1994.

[44] H. Cheng, X. Jiang, Y. Sun, and J. Wang, "Color Image Segmentation: Aadvances and Prospects," *Pattern Recognition*, vol. 34, no. 12, pp. 2259–2281, 2001.

[45] L. Lucchese and S. Mitra, "Color Image Segmentation: A State-of-the-Art Survey," *Proceedings of the Indian National Science Academy*, vol. 67 A, pp. 207–221, 2001, New Delhi, India.

[46] S. Shafer, "Using Color to Separate Reflection Components," *Color Research and Application*, vol. 10, no. 4, pp. 210–218, 1885.

[47] G. Healey, "Segmenting Images Using Normalized Color," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 22, pp. 64–73, January 1992.

[48] B. A. Maxwell and S. A. Shafer, "Physics-Based Segmentation of Complex Objects Using Multiple Hypotheses of Image Formation," *Computer Vision and Image Understanding*, vol. 65, no. 2, pp. 269–295, 1997.

[49] J. B. MacQueen, "Some Methods for Classification and Analysis of Multivariate Observations," *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, pp. 281–297, 1967, University of California, Berkeley.

[50] D. Comaniciu and P. Meer, "Robust Analysis of Feature Spaces: Color Image Segmentation," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 750–755, June 1997, San Juan, Puerto Rico.

[51] T. Pappas, "An Adaptive Clustering Algorithm for Image Segmentation," *IEEE Transactions on Signal Processing*, vol. 40, no. 4, pp. 901–914, April 1992.

[52] E. Saber, A. M. Tekalp, and G. Bozdagi, "Fusion of Color and Edge Information for Improved Segmentation and Edge Linking," *Image and Vision Computing*, vol. 15, no. 10, pp. 769–780, 1997.

[53] M. Celenk and M. U. de Haag, "Optimal Thresholding for Color Images," *Proceedings of the SPIE - The International Society for Optical Engineering, Nonlinear Image Processing*, vol. 15, no. 10, pp. 250–259, January 1998, San Jose, CA.

[54] D.-C. Tseng, Y.-F. Li, and C.-T. Tung, "Circular Histogram Thresholding for Color Image Segmentation," *Proceedings of the 3rd International Conference on Document Analysis and Recognition*, vol. 2, pp. 673–676, August 1995, Montreal, Canada.

[55] K. Sobottka and I. Pitas, "Segmentation and Tracking of Faces in Color Images," *Proceedings of the 2nd International Conference on Automatic Face and Gesture Recognition*, pp. 236–, 1996, Killington, VT.

[56] L. Lucchese and S. Mitra, "Unsupervised Segmentation of Color Images Based on k-means Clustering in the Chromaticity Plane," *Proceedings of the IEEE Workshop on Content-Based Access of Image and Video Libraries*, vol. 0, p. 74, 1999, Fort Collins, CO.

[57] G. R. Cross and A. K. Jain, "Markov Random Field Texture Models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-5, no. 1, pp. 25–39, January 1983.

[58] D. K. Panjwani and G. Healey, "Markov Random Field Models for Unsupervised Segmentation of Textured Color Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, pp. 939–954, 1995.

[59] M. Celenk, "Hierarchical Color Clustering for Segmentation of Textured Images," *Proceedings of the 29th Southeastern Symposium on System Theory*, pp. 483–487, March 1997, Cookeville, TN.

[60] S. Ji and H. Park, "Image Segmentation of Color Image Based on Region Coherency," *Proceedings of the IEEE International Conference on Image Processing*, vol. 1, pp. 80–83, October 1998, Chicago, IL.

[61] A. Round, A. Duller, and P. Fish, "Colour Segmentation for Lesion Classification," *Proceedings of the 19th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, vol. 2, pp. 582–585, November 1997, Chicago, IL.

[62] A. Tremeau and N. Borel, "A Region Growing and Merging Algorithm to Color Segmentation," *Pattern Recognition*, vol. 30, no. 7, pp. 1191–1203, 1997.

[63] Y. Kanai, "Image Segmentation Using Intensity and Color Information," *Proceedings of the SPIE The International Society for Optical Engineering, Visual Communications and Image Processing*, pp. 709–720, January 1998.

[64] Y. Deng, B. Manjunath, and H. Shin, "Color Image Segmentation," *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, vol. 2, p. 2446, June 1999, Fort Collins, CO.

[65] P. Campadelli, D. Medici, and R. Schettini, "Color Image Segmentation Using Hopfield Networks," *Image and Vision Computing*, vol. 15, no. 3, pp. 161 – 166, 1997.

[66] N. Funakubo, "Feature Extraction of Color Texture Using Neural Networks for Region Segmentation," *Proceedings of the 20th Annual Conference of IEEE Industrial Electronics*, vol. 2, pp. 852–856, September 1994, Bologna, Italy.

[67] T. Carron and P. Lambert, "Color Edge Detector Using Jointly Hue, Saturation and Intensity," *Proceedings of the IEEE International Conference on Image Processing*, vol. 3, pp. 977–981, November 1994, Austin, TX.

[68] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active Contour Models," *International Journal of Computer Vision*, vol. 1, pp. 321–331, 1988.

[69] D. Mumford and J. Shah, "Boundary Detection by Minimizing Functionals," *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 22–26, June 1985, San Francisco, CA.

[70] D.-C. Tseng and C.-H. Chang, "Color Segmentation Using Perceptual Attributes," *Proceedings of the 11th IAPR International Conference on Pattern Recognition*, vol. 3, pp. 228–231, September 1992, La Haye, Holland.

[71] J. Brand and J. Mason, "Skin Probability Map and Its Use in Face Detection," *Proceedings of the IEEE International Conference on Image Processing*, vol. 1, pp. 1034–1037, 2001, Thessaloniki, Greece.

[72] Z. Xue, D. Shen, and S. Wong, "Tissue Probability Map Constrained CLASSIC for Increased Accuracy and Robustness in Serial Image Segmentation," *Proceedings of the 2009 SPIE Symposium on Medical Imaging*, vol. 7258, pp. 725 904–725 904–9, February 2009, Lake Buena Vista, FL.

[73] B. Momjian, *PostgreSQL: Introduction and Concepts.* Boston, MA: Addison-Wesley Longman Publishing Co., Inc., 2001.

[74] J. D. Touch, "Performance Analysis of MD5," *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, pp. 77–86, October 1995, Cambridge, MA.

[75] R. P. Brent, "Fast Multiple-Precision Evaluation of Elementary Functions," *Journal of the ACM*, vol. 23, pp. 242–251, April 1976.

APPENDICES

A. IMAGE ACQUISITION PROTOCOL

This is the protocol we used for acquiring test images for the GARI database. The images are used for testing various functions of the GAR system.

- Persons involved
  - 2 GARI staff members
  - 1 or more persons from Police Department
- Equipment/Materials needed
  - Pens or pencils
  - 2 Digital Camera (1MPx and above)
  - 2 Tripods
  - 2 Mobile Telephone with Android OS
    - Built-in camera (1MPx and above)
    - GPS receiver
    - optional: Data plan
  - 1 GPS receiver
  - Graffiti Information Forms
  - Fiducial Markers
  - Image Checklist
  - 1 Purdue University owned laptop
  - 1 External hard drive

1) Preliminaries (Internet connection required)
   a) Check time setting on the two Android mobile telephones, the two digital cameras, and the GPS receiver using the Purdue University owned laptop, and ensure they are in sync with the GARI server.

b) Make sure the two Android mobile telephones, the two digital cameras, and the GPS receiver batteries are fully charged.

c) Verify all equipment/materials above are available.

d) Make sure the settings of the two digital cameras are set to default by finding the appropriate menu option.

e) Turn flash feature off on the two Android mobile telephones built-in cameras and the two digital cameras.

f) Make sure zoom and macro features are not enabled on the two Android mobile telephones built-in cameras and the two digital cameras.

g) Assign each person an ID number, and record it on the Graffiti Information Form.

h) Record person's name and affiliation on the Graffiti Information Form.

2) Set up environment

a) Stand up in front of the graffiti, far enough so that the cameras can capture all the content, preferably perpendicular to the surface containing the graffiti. Some angle margin is permitted (θ spherical degrees), as shown in Figure A.1 and Figure A.1. This angle show be small enough so that the graffiti contents can be identified properly.

b) Make sure weather condition does not prevent seeing the graffiti.

c) Place the fiducial marker in a spot that would be 20 inches away and parallel to the surface containing the graffiti, as shown in Figure A.1 and Figure A.2. It should not block the graffiti contents.

d) Make sure there are not any objects between the camera and the graffiti that obstruct partially or totally the view of the graffiti.

e) Record Date (MM/DD/YYYY), Time (HH:MM:SS) and GPS coordinates (latitude, longitude and altitude, with six digit precision) on the Graffiti Information Form. Obtain the information from the GPS receiver.

f) Record neighborhood description on the Graffiti Information Form. Specify street name(s) and landmarks in the area near the graffiti.

g) Proceed to take image. For each graffiti, take six images, using

- Android mobile telephone 1
- Android mobile telephone 1
- Android mobile telephone 2
- Digital camera 1 with tripod
- Digital camera 1 without tripod
- Digital camera 2 with tripod
- Digital camera 2 without tripod

h) For each graffiti, record the device(s) used on the Graffiti Information Form.

3) Taking an image of a graffiti

3.1) Taking image of a graffiti using an Android mobile telephone

a) Launch GARI application on the Android mobile telephone and assign an Image Taker ID, corresponding to the one assigned in step 1. Preliminaries.

b) Select the "Capture Image" option from the GARI application main menu. The camera activity is then initialized.

c) Prepare for taking the image (position of the camera as desired, within the recommended distance and angle from the graffiti). Make sure all the contents of the graffiti and the entire fiducial marker can be seen on the device screen.

d) Take an image of the graffiti, trying to maintain the device's position, as much as possible.

e) If the image does not meet the requirements noted in the Image Checklist, the image should be retaken.

f) If location available through WiFi/GSM/GPS the GPS coordinates will be automatically stored in the image. If no location method available, will receive a message: "No NETWORK/GPS found. Check coordinates manually!". Ignore it, since the GPS coordinates have already been recorded on the Graffiti Information Form.

g) Crop the image if desired.

h) Select the "Send to Server" option from the GARI application main menu. If no Internet connection available, will receive a message: "No internet connection available". It means the image has not been uploaded to the server. However, the image is still in the Android mobile telephone SD card, and it can be copied to a computer at the end of the session (Section 5.a of the protocol), and uploaded in the future. If the image has not been uploaded to the server, check the box "Not Successfully Uploaded" on the Graffiti Information Form.

3.2) Taking image of a graffiti using a digital camera

a) If a tripod is used, attached it to the digital camera, and adjust it so the digital camera is at the same position as if it is held without using the tripod.

b) Prepare for taking the image (position of the camera as desired, within the recommended distance and angle from the graffiti). Make sure all the contents of the graffiti and the entire fiducial marker can be seen on the device screen.

c) Take an image of the graffiti, trying to maintain the device's position, as much as possible.

4) Completing the Graffiti Information Form

a) Fill the "Ground-truth graffiti information" section on the Graffiti Information Form with ground-truth information associated with the graffiti, if known. It includes:

- Graffiti color(s): color or colors of the graffiti contents.

- Gang Name(s): name of the gang or gangs that participated on the drawing of the graffiti.
- Gang Member(s): name of the gang member or gang members that participated on the drawing of the graffiti.
- Target Gang Name(s): name of the gang or gangs that are targeted in the graffiti.
- Target Gang Member(s): name of the gang member or gang members that are targeted in the graffiti.
- Symbol(s): description of the symbol(s) in the graffiti, including color, position in the graffiti (e.g. next to the gang name), orientation (e.g. upside down fork), and possible meaning.
- Other content(s): description of other relevant contents of the graffiti (e.g. crossed letters, nicknames), including color, position in the graffiti (e.g. crossed C on the right of BERO), and possible meaning.
- Comments: additional information of the graffiti that does not fit in the previous subsections of the "Ground-truth graffiti information" section.

b) Fill the "General Comments" section on the *Graffiti Information Form* with additional comments that do not fit in all the previous sections.

5) End of the session procedures

a) Copy all the images taken with the Android mobile telephones (stored in the GARI folder) and with the two digital cameras to a Purdue University owned laptop and to an external hard drive.

b) Take cards out of the digital cameras and reformat them.

c) Ensure the Purdue University owned laptop and the two digital cameras are synced.

d) Recharge laptop and camera batteries.

e) Store fiducial markers and other materials in a safe place for later use.

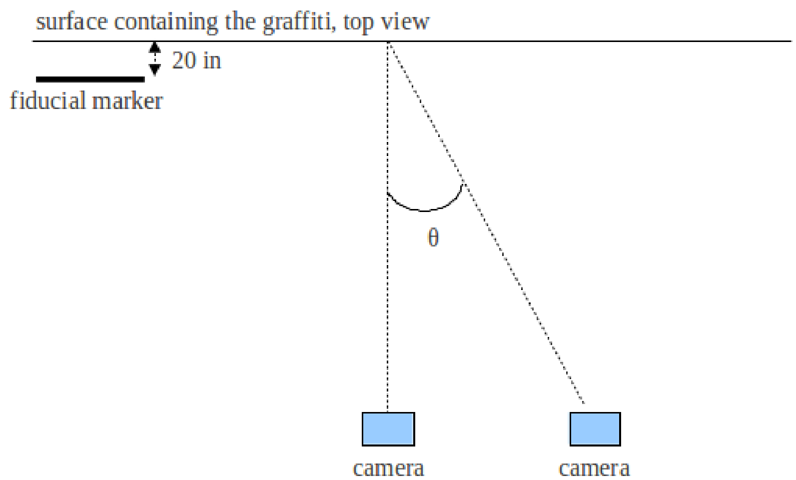


Fig. A.1.: Top view of the setup environment.

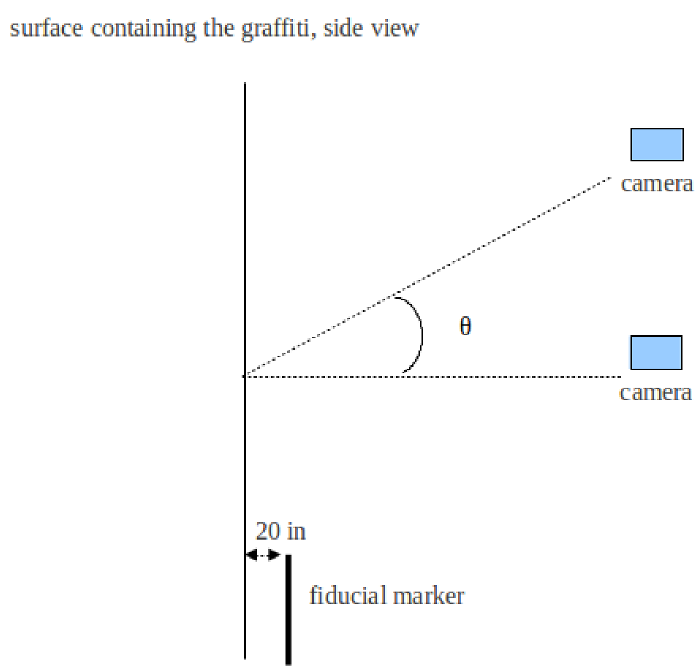


Fig. A.2.: Side view of the setup environment.

Graffiti Information Form

First Responder information

Name: ID: Affiliation:

Basic image information

Date: / / Time: : :

GPS Coordinates

Latitude: Longitude: Altitude:

Neighborhood Description:

Device(s) used

☐ Android Mobile Telephone ☐ Camera with tripod ☐ Camera without tripod

---

Ground-truth graffiti information

Color(s):

Gang Name(s):

Gang Member(s):

Target Gang Name(s):

Target Gang Member(s):

Symbol(s):

Other Content(s):

Comments:

General Comments:

Fig. A.3.: Graffiti Information Form.

B. DATABASE TABLES

This Appendix describes the database tables in more detail.

Table B.1: EXIF data fields in table *images* of the gang graffiti database.

EXIF field	**Description**
filesize	Size of the image (bytes)
filedatetime	Date and time of capture
resolutionheight	Height of image (px)
resolutionwidth	Width of image (px)
focallength	Focal Length of camera's optical system
isoequiv	ISO equivalent value used
cameramake	Camera make
cameramodel	Camera model
gpsaltitude	GPS altitude
gpslongitude	GPS longitude
gpslatitude	GPS latitude
xresolution	DPI in the width direction
yresolution	DPI in the height direction
ycbcrpositioning	Position of the YCbCr components
fnumber	F number
compressedbitsperpixel	Compressed bits per pixel
exposuretime	Exposure time (seconds)
exposurebias	Exposure bias (APEX)
aperture	Lens aperture (APEX)

meteringmode	Metering mode
flash	Status of flash when the image was shot
interoperabilityoffset	Interoperability offset
sensingmethod	Sensing method
customrendered	Use of special processing on image data
whitebalance	White balance
digitalzoomratio	Digital zoom ratio
exposuremode	Exposure mode

Table B.2: Image location fields in table *images* of the gang graffiti database.

Field	**Description**
country	Country (given GPS coordinates)
state	State (given GPS coordinates)
county	County (given GPS coordinates)
city	City (given GPS coordinates)
zip	ZIP code (given GPS coordinates)
address	Address (given GPS coordinates)

Table B.3: Graffiti analysis fields in table *images* of the gang graffiti database.

Field	**Description**
gangnameia	Gang name from IA [1]
gangnamegt	Gang name from GT [2]
gangidia	Gang ID from IA

[1] IA: Image Analysis
[2] GT: Ground Truth

gangidgt	Gang ID from GT
gangmembernameia	Gang member name from IA
gangmembernamegt	Gang member name from GT
gangmemberidia	Gang member ID from IA
gangmemberidgt	Gang member ID from GT

Table B.4: Image information fields in table *images* of the gang graffiti database.

Field	**Description**
imageid	Image ID
path	Path to the image file
firstrespondername	First responder name
firstresponderid	First responder ID
comment	Comments about graffiti
webupload	File uploaded from desktop version (boolean)
realcoords	Image has real GPS coordinates (boolean)
filedatetimeupload	Date and time the file was uploaded to the database
lastmodified	Date and time a fields was last modified
lastmodifiedname	First responder that last modified a field

Table B.5: User information fields in table *users* of the gang graffiti database.

Field	**Description**
id	User ID
password	MD5 hash of user's password
name	User's name

admin	User is administration (boolean)
first	First login (boolean)
gmail	Gmail address
email	Alternative email address
affiliation	User affiliation
android	Has Android application (boolean)
comments	Comments about user

Table B.6: Image blobs information fields in table *imageBlobs* of the gang graffiti database.

Field	**Description**
imageid	Image ID
blobid	Blob ID for a particular image ID
componentid	Component ID for a particular blob ID
colorid	Color ID for a particular component ID
crossedout	Boolean to determine if the component is crossed-out
upsidedown	Boolean to determine if the component is upside-out