iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 7 **Issue:** III **Month of publication:** March 2019

DOI: http://doi.org/10.22214/ijraset.2019.3081

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 7 Issue III, Mar 2019- Available at www.ijraset.com

An Approach Implementing Mobile Application for Herbal Treatment

Dr. S. Hemalatha[1], Nidharsana M[2], Deebika kumari T[3], Priyanka K[4]
[1]*Professor, Department of Computer Science, Panimalar Institute Of Technology*
[2, 3, 4]*Final year, Department Of Computer Science, Panimalar institute of Technology*

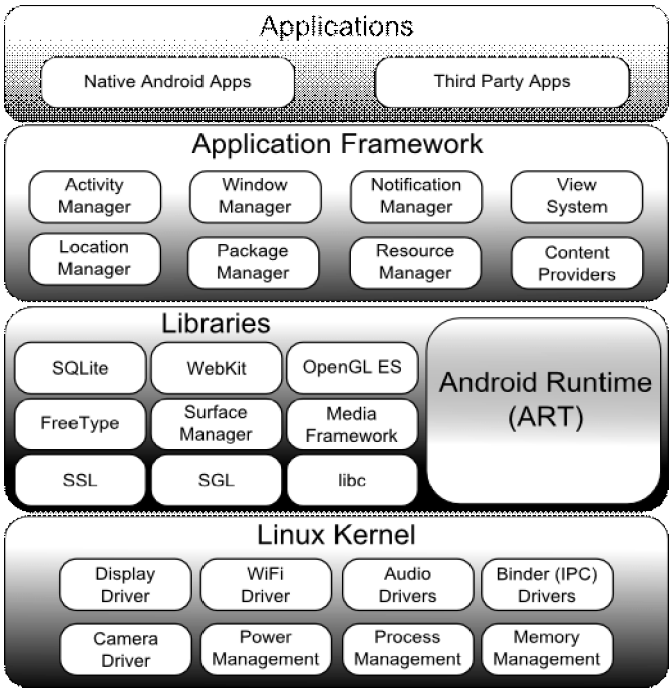
Abstract: ***A fully automated method for the recognition of medicinal plants using computer vision and machine learning techniques has been presented. Leaves from 24 different medicinal plant species were collected and photographed using a smart phone in a laboratory setting. A large number of features were extracted from each leaf such as its length, width, perimeter, and area, number of vertices, colour, perimeter and area of hull. Several derived features were then computed from these attributes. The best results were obtained from a SVM classifier using a 10-fold cross-validation technique. With an accuracy of 90.1%, SVM classifier performed better than other machine learning approaches such as the k-nearest neighbour, Naïve Bayes, KNN and neural networks. This is implemented using Android studio software and we obtained a working module.***

Keywords: leaf recognition, noise removal, SVM classifier, Android application, Feature extraction.

I. INTRODUCTION

Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on Jet Brains IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems. It is a replacement for the Eclipse Android Development Tools (ADT) as the primary IDE for native Android application development. Android Studio was announced on May 16, 2013 at the Google I/O conference. It was in early access preview stage starting from version 0.1 in May 2013, then entered beta stage starting from version 0.8 which was released in June 2014. The first stable build was released in December 2014, starting from version 1.0. The current stable version is 3.3, which was released in January 2019.

II. ARCHITECTURE OF ANDROID



A. Advantages Of Android

Use a Different Messaging App for SMS Android Offers an Open Platform Easy access to the Android App Market Cost Effective

B. Disadvantages Of Android

Usually you need more code on Java than Objective- C. Complex layouts and animations are harder to code in Android.
Applications contains virus also present in Android

1) Market

a) A lot of "process" in the background that lead to the battery quickly drains.
b) Advertise, will always be ads on display, either the top or bottom of the application.
c) Low security and fake apps can be installed to steal your info from unknown resources
d) High device fragmentation

III. SURVEY

Oscar Somarriba[1] Smart devices are everywhere nowadays, such as smartphones and tablets where the Android platform is dominant in this mobile era. As a consequence of this popularity, the malware targeting Android smartphones has also mushroomed. Android malware is one of the major security issues and fast growing threats facing the Internet in the mobile arena, today. So, in this context, DNS (Domain Name System) is widely misused by miscreants in order to provide internet connection within malicious networks and botnets. In our experiments, we use the MalGenome dataset in order to generate network traffic. Besides, most of the malware we examine use DNS in order to obtain the IP address of their command and control servers. Then, the problem of determining the DNS queries done by the malware through devices without modifying the firmware or rooting smartphone, is very important and it poses a big challenge. From traces we generated from apps under test, we can extract malicious URLs invoked by the malware.

Shun Kurihara, Shoki Fukuda, Saneyasu yamaguchi, Ayano Koyanagi, Masato Oguchi, Ayumu Kubota, Akihiro Nakarai [2] Android OS has a function with which an application can work in screen-off state without user's operation. In this paper, we propose a method for identifying applications which largely drain battery in Screen-off state in Android devices. We monitor the wake-up of Android devices and estimate the power consumption of each application based on the monitoring results. Our experimental results demonstrate that our method can identify power draining applications effectively.

Chein Hung Liu, Chien Yu Lu, Shan Jen Cheng, Koan Yuh Chang, Yung Chia Hsiao, Weng Ming Chu[3] With the widespread popularity of Android devices, the number of Android applications has increased dramatically in recent years. In order to assure the quality of the applications, Android testing has drawn extensive attention. This paper proposes an approach to automate the testing of Android applications based on the Capture and Replay method. Particularly, the user events of Android applications are captured and converted into Robotium test scripts that can be executed to replay the recorded actions of users. The approach also allows inserting assertions when capturing user interactions for verifying the outputs of Android UI components. A supporting tool is implemented to illustrate the usefulness of the proposed approach.

Cangzhou Yuan, Shenhong Wei, Yutong Wang, Yue You, Shang Guan Ziliang[4] The rapid growing of the Android application and malware has increased the usage of the application category in Android malware detection and application searching. However, the defects of management of Android Market lead to a great deal of applications miscategorization. Therefore, it's helpful for both organizing the Android Market and Android malware detection to give an approach that can automatically distinguish different categories of the applications. In this paper, we present an effective approach for automatically categorizing Android applications based on Bayesian classification. Considering the category of the application is determined by its function, we extracted the used permissions and strings that can reflect the application function from the application itself and Android Market as classification features. Finally, we conduct experiments with 13005 applications that are composed of 18 categories with Naive Bayes. The evaluation results show that our approach can achieve better accuracy and performance than previous coarse-grained feature extraction methods.

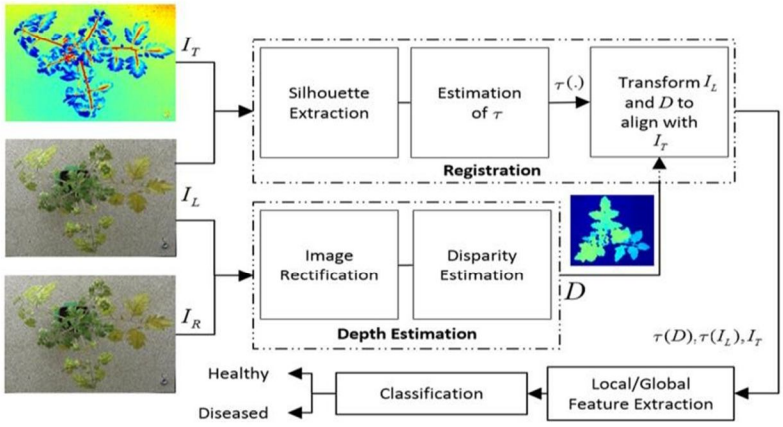
K.Kavitha, P.Salini, V.Ilamathy[5] Android plays a vital role in the today's market. According to recent survey placed nearly 84.4% of people stick to android which explosively become popular for personal or business purposes. It is no doubt that the application are extremely familiar in the market for their amazing features and the wonderful benefits of android applications makes the users to fall for it. Android imparts significant responsibility to application developers for designing the application with understanding the risk of security issues. When concerned about security, malware protection is a major issue in which android has been a major target of malicious applications. In android based applications

permission control is one of the major security mechanisms. In this project, the permission induced risk in application, and the fundamentals of the android security architecture are explored, and it also focuses on the security ranking algorithms that are unique to specific applications. Hence, we propose the system providing the detection of malware analysis based on permission and steps to mitigate from accessing unwanted permission (limits the permission). It is also designed to reduce the probability of vulnerable attacks.

IV. IMPLEMENTATION BLOCK DIAGRAM

Once the image is captured the initial step will be a pre- processing step, it will process the image and removes voices to enhance it. It also includes enhancing the brightness for dim images. Silhouette extraction step enhances the shadow image to enhanced image by calculating the torque value and matches it with image that have high clarity. The leaves are classified as healthy and unhealthy leaves and their features are also extracted. This may train the system further better.

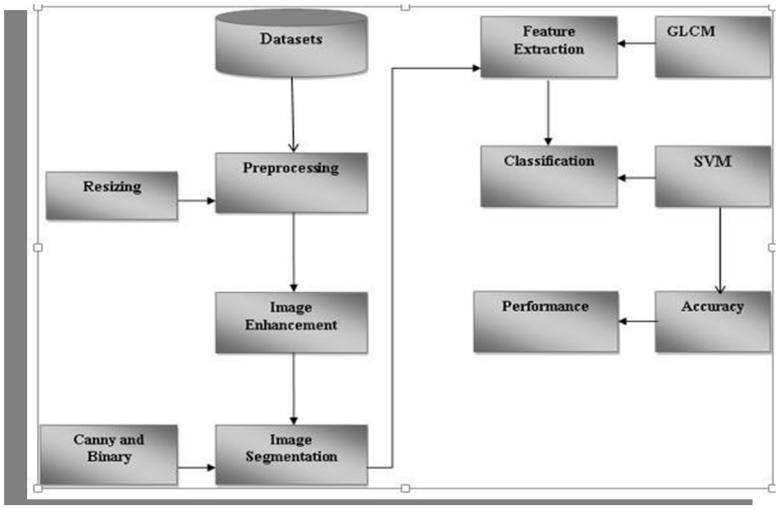
I Module

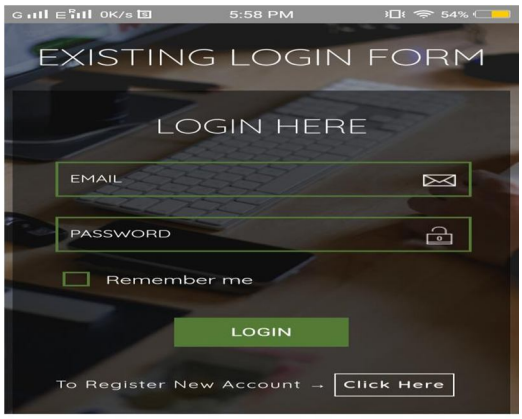


V. SNAPSHOTS

This is the overall block diagram of our project. Here the datasets consists of plants images and descriptions. All the functions that requires datasets can access this module. From the datasets preprocessing step is done to remove noise, enhancement module is used to enhance the color or shadow removal of the image. Segmentation splits the datasets into parts to check the quality of the image. After segmentation features from each segment is extracted and analyzed using GLCM. And finally the performance and accuracy are noted to make future enhancements.

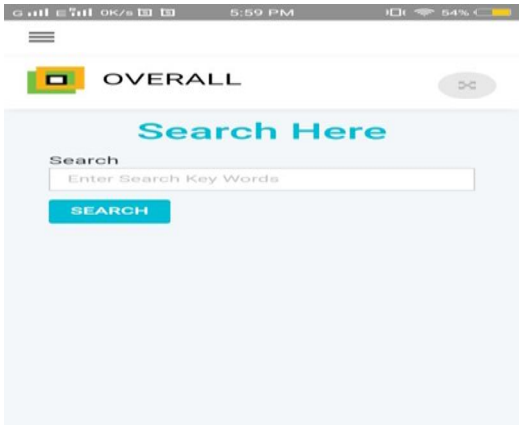
A. Architecture Diagram





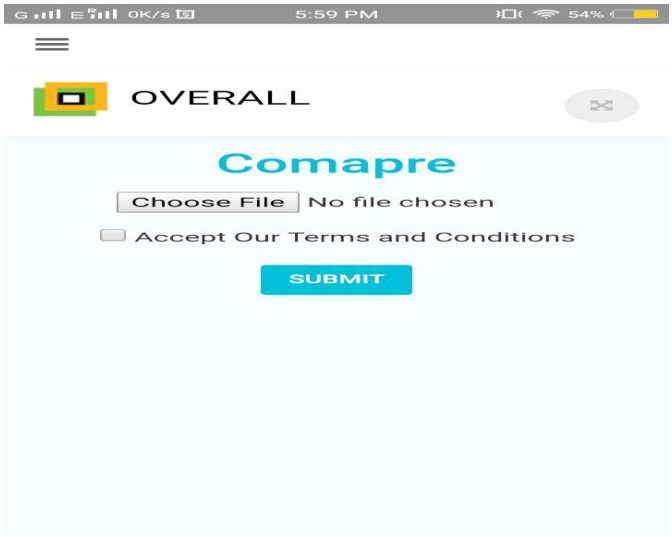
In this module the user have to create a login for them by using unique mail id and password. Once if they registered they can login with this details at any time.

II Module



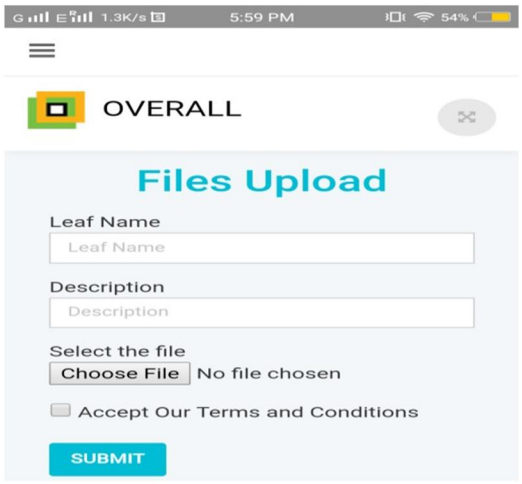
In this module we have attached a search bar, where the user can feed their symptoms as input. In the backend this symptom is compared with the descriptions in our dataset, and the correct medicinal leaf is suggested to consume.

III Module



If the user wants to check whether they are consuming the correct medicinal plant. They can scan their sample and compare it with the samples which we gave as datasets. If both the samples matches it will provide correct result. In the backend noise removal process takes place which assures you the high accuracy on your results.

IV Module



If the user is willing to add new datasets in our app, they can use this add new feature to upload their datasets including descriptions and images. The description doesn't have any limits; it can be of any number of lines.

VI. CONCLUSION

Thus by using this application user can able to treat themselves using the medicinal plants which is suggested. Hence they don't need to visit the health care and wait in a queue to get treated. The future enhancements may be done as, if the symptoms doesn't cure for duration which is already provided, a high dosage plant can be suggested as per requirement.

REFERENCES

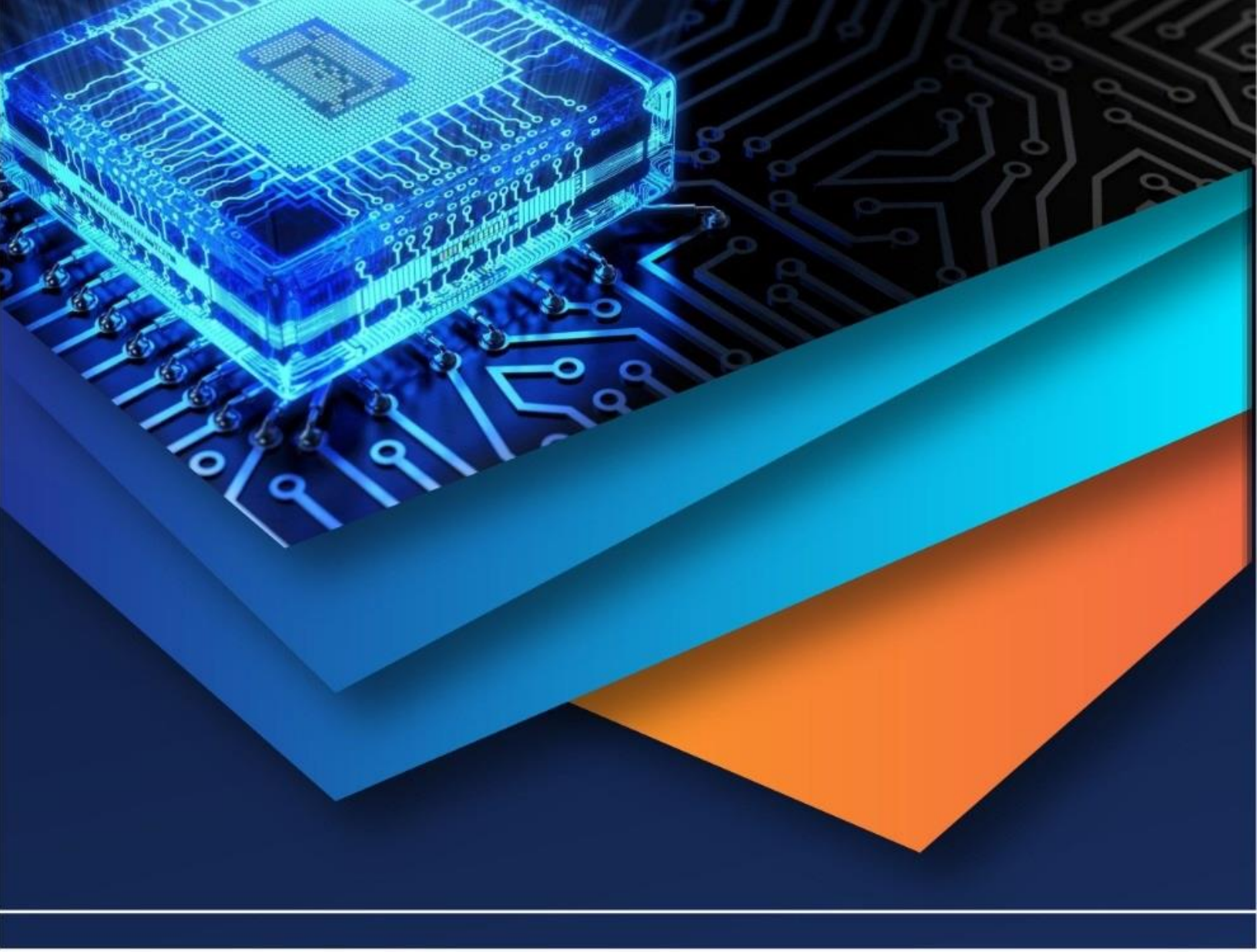
[1] Oscar Somarriba, "Detecting blacklisted URL's from unmodified and non-rooted android devices" in 2017 IEEE 37th Central America and panama convention (CONCAPAN XXXVII) 15-17 Nov 2017

[2] Shun Kurihara, Shoki Fukuda, Saneyasu yamaguchi, Ayano Koyanagi, Masato Oguchi, Ayumu Kubota, Akihiro Nakarai, "A study of identifying battery-draining android applications in screen-off state" in 2015 IEEE 4th global conference on consumer electronics (GCCE) 27-30 Oct 2015.

[3] Chein Hung Liu, Chien Yu Lu, Shan Jen Cheng, Koan Yuh Chang, Yung Chia Hsiao, Weng Ming Chu, "Capture- replay testing for android applications" in 2014 International symposium on computer, consumer and control, 10-12 June 2014.

[4] Cangzhou Yuan, Shenhong Wei, Yutong Wang, Yue You, Shang Guan Ziliang, "android applications categorization using Bayesian classification" in 2016 International conference on cyber-enabled distributed computing and knowledge discovery(CyberC) 13-15 Oct 2016.

[5] K.Kavitha, P.Salini, V.Ilamathy, "Exploring the malicious android applications and reducing risk using static analysis" in 2016 International conference on Electrical, Electronics, and optimization techniques (ICEEOT) 3-5 March 2016.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY