

# Real-Time Tree Counting Android Application and Central Monitoring System

**Ahmet Ali Süzen**

Department of Computer Technologies, Isparta University of Applied Sciences, Isparta, Turkey  
E-mail: ahmetsuzen@isparta.edu.tr

**Remzi Gürfidan**

Department of Computer Technologies, Isparta University of Applied Sciences, Isparta, Turkey  
E-mail : remzigurfidan@isparta.edu.tr

**Kıyas Kayaalp**

Department of Computer Technologies, Isparta University of Applied Sciences, Isparta, Turkey  
E-mail : kiyaskayaalp@isparta.edu.tr

**Mehmet Ali Şimşek**

Department of Computer Technologies, Tekirdağ Namik Kemal University, Tekirdağ, Turkey  
E-mail : masimsek@nku.edu.tr

Received: 18 November 2019; Accepted: 25 December 2019; Published: 08 April 2020

**Abstract**—In this study, a cloud-based android application and centralized tracking software were developed to perform an accurate and uninterrupted tree count across open lands. The application is used to count the desired number of trees and species at the same time. User-logged data and location information are saved in real-time to the application's cloud database. The application can work online and with offline mod. In cases where there is no internet connection, it inserts the data to the local SQLite database. After the connection is established, the pairing continues. It's used Google Firebase on the cloud server for data storage. The processing of target locations and GPS coordinates was developed with the Google Map Library. The tree counting application automatically picks up the user's current location when it is first opened. The counting starts after the tree and tree species that the user has selected from the menu. The software developed shows that tree counting is done simultaneously at the desired point. It also solves confusion caused by different tree species during the counting. We've received feedback from 100 people using the application. The users answered five questions. As a result, it is aimed to provide a comfortable transition between tree species and its users with its simple use to eliminate the complexity of counting and save time.

**Index Terms**—Android, Cloud System, Firebase, Location based tracking, Tree count.

## I. INTRODUCTION

Many natural disasters in agriculture and the risks of

climate change are important factors. These risks have been reduced by the state in such a way as to reduce the burden of both the agricultural sector and the farmer. The system that reduces this burden is also called agricultural insurance in the draft law on Agricultural Insurance [1]. About 5.5 million people in Turkey earn their living from farming, which depends on the agricultural sector. These people protect their annual livelihoods by having their existing products insured against natural disasters and adverse climatic conditions [2]. The most important part of the insurance process is to be able to accurately determine the number of trees in the farmer's Garden [3]. This data is of great importance both in terms of insurance costs and in terms of farmers' crop provision. The process of counting products for accurate insuring has huge difficulty in the field. These difficulties can be sorted as follows.

- Having more than one crop variety in the land.
- The number of trees in the large land is tens of thousands
- In the carelessness, the counting starts over and big time losses occur.

These problems are partially solved by the calculator or counting applications on the user's phone. However, the long-distance between the trees and the variety of trees continues problems. When the studies in the literature are examined, it is seen that tree counting is done with real-time image processing [4,5] and artificial intelligence techniques [6,7]. For these studies to achieve high accuracy, the data must be in the tree to be detected earlier. Images must be taken with drones before and counting must take place via drones for counting [8].

Although they are related to the existing problem, they cannot be used due to high cost and applicability. Solving problems in tree counts will be solved with applications developed for this problem.

A location-based Android application has been developed to solve the problems insurance companies face in tree counting. The application has two purposes. The first is that users can perform the correct tree count on open land. The second purpose is to ensure that the location of the census is followed by the center. At the same time, the user's data has been stored and made available for reporting by cloud computing technology. As a result, more practical and safe tree counting will be provided with the use of the application.

## II. RELATED WORKS

Tree counting is one of the topics studied in the literature for the last 10 years in order to make better financial studies such as product productivity, earnings, and cost per tree. Tree counting continues to be tried on different techniques and different trees. Some of the studies in the literature with tree counting are summarized in the following section.

Tree counts are often performed by processing images taken by satellite images or drones. OLICOUNT software developed for the counting of olive trees is used with the pictures captured from the satellite [9]. For the pistachio tree counting, the outer boundaries of the tree were first set. Then, the images taken from the satellite were divided into RGB bands with the ArcGIS program and the tree counting process was attempted to be done [10]. Tree counting was performed by a four-step algorithm (asymmetric smoothing filter, local minimum filter, mask layer, and spatial aggregation operator) from very high-resolution satellite data [11]. Tree counting was carried out using the level-set method for counting palm trees from images obtained from drones [12]. In another proposed method for counting palm trees, the convolutional neural network is trained and optimized. It has been stated that 96% accuracy rate has been reached in tree counting [13].

## III. SYSTEM ARCHITECTURE

The study developed can run on mobile phones or tablets. The user uses the application in an external environment. Therefore, data must be collected centrally. This is why data storage is done on cloud servers. Cloud technology is a service that does not require installation and provides the opportunity to store the data of web-based applications online [14]. Because multiple backup areas are used on cloud systems, reliability is increased [15]. One of the most useful examples of this is Firebase. Developed by Google, Firebase is a database that allows users to store data under different conditions. It is preferred because of its high performance and efficiency. There is no need to write server-side code in Firebase technology [16]. The Google firebase database basically

offers the following features [17,18].

- Real-time database
- User login authorization
- Storage
- Machine learning kit
- Performance testing environment
- Cross-Platform Application Analysis
- Common function
- Advertising materials
- Notification submission
- General application management

Firebase database application development occurs in different situations. These are Real-time Database, Authentication, Test Lab, Crashlytic, and Firestore, respectively [19,20].

Although the study is a mobile application, there is desktop software in the center for auditing purposes. This software was developed for the purpose of analyzing data and coordinates from the mobile application. The software was developed using the C# 6.0 programming language with visual studio 2017. 3-Tier architecture (database, connection and interface) was also used. The desktop software allows a person from the center to monitor while counting on open terrain. In this way, counting is prevented without going to the counting point.

The mobile application runs on phones and tablets that have Android 5.0 operating systems and over. When the application is first opened, it requires an internet connection to transmit location and count data. The application will run in offline mode if you do not have an internet connection at that moment. In this mode, the data will be temporarily saved to the SQLite database. The table in which the data is stored in offline mode in the mobile application is shown in Fig. 1. When the Internet connection is established, it will be paired with the cloud system.


OfflineMode	
	Id
	Location
	PersonId
	TreeCount
	TreeType
	TreeCountInfo

Fig.1. SQLite offline mode table

If the application is running in online mode, the counting process (increment or decrement) and the current location are saved in the database on the server. This ensures location accuracy in the counting. The architecture of the developed application is given in Fig.2.

Mobile counting application is developed as native because it will use the hardware features of mobile phones and tablets. Therefore, the development process was used with Android Studio. Firstly, the permission definition has been made for the mobile application. This

permission is required to use the user's location and internet information. The coarse-code for the permissions received for the application are given Table 1.

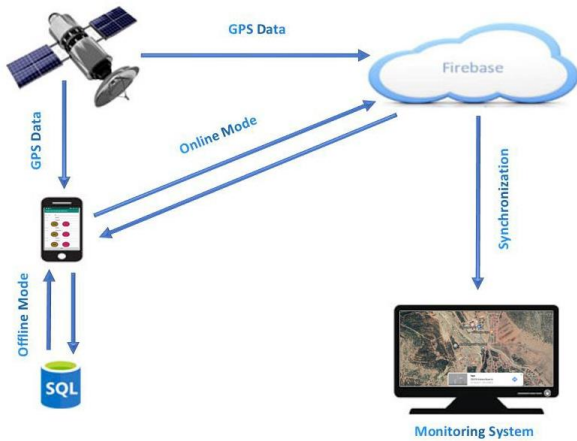


Fig.2. System structure

Table 1. Permissions and descriptions used in the application

Permission	Description
INTERNET	using Internet
ACCESS_COARSE_LOCATION	Receive location information from GPS
ACCESS_FINE_LOCATION	Receive location information from GSM or WIFI
MAPS_RECEIVE	Map service
ACCESS_NETWORK_STATE	Connection status

In a second preparation, Google services need to be added to the project to transfer real-time data to the firebase database. For these configurations; It needs to be added "implementation 'com.google.firebase:firebase-database:11.8.0'" into of Build.gradle(Module:app) file and "classpath 'com.google.gms:google-services:4.0.1'" to "Build. Gradle (Project :Proje Android)" file.

When the application is opened for the first time, it is determined by *Internet\_Control ()* function, which will run in online or offline mode. This function listens the *CONNECTIVITY\_SERVICE* structure with the *ConnectivityManager* class. From this service, *getActiveNetworkInfo()* function determines if there is an internet connection. If there is internet, online mode is selected, otherwise offline mode is selected.

Moment coordinate information is sent to the server to determine that the counting is done from the correct location. *Android.location.Location* library is used to access GPS information. Within this library, the *handleLocationUpdate()* function and the location information come as a string. GPS data is received in the following format. *ParseLatitude ()* and *parseLongitude ()* functions are used to obtain latitude and longitude information from the collected GPS data. Thus, the desired location information is received.

\$GPRMC,<Time>,<Status>,<Latitude>,<Longitude>,<Speed>,<Angle>,<Date>,<Variation>,<Integrity>,<Checksum>

In practice, the data is stored in two different ways. It is stored to the SQLite database in offline mode using the *SQLiteDatabase* class. The entered data is sent to the database via the *ContentValues* object with the *getWritableDatabase* function. An *Internet\_Control()* function runs every minute and the application switches to online mode if there is the internet. Then the data stored in the SQLite database is transferred to the cloud system with the *Transfer()* function. In online mode, data is sent using the *setValue ()* function with the *FirestoreDatabase* Library.

The operating structure and the rough algorithm of the application in online and offline mode are shown in Fig. 3.

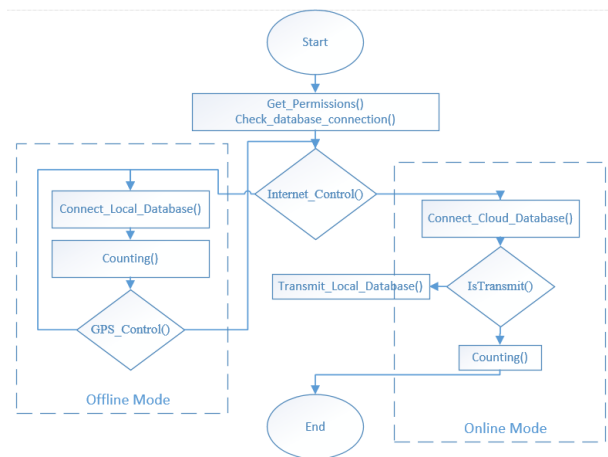


Fig.3. Application algorithm

IV. APPLICATION TESTING

There are three main menus in the application performed. By selecting the "Starting New Count" menu, a new count section is opened in the cloud base. We continue to process the data of the last count, assuming that there is a break or a deceleration during the last count with the "Continue to The Last Count" menu. "Previous Counts" is the menu where the previous counts are reported and stored and listed (Fig. 4).

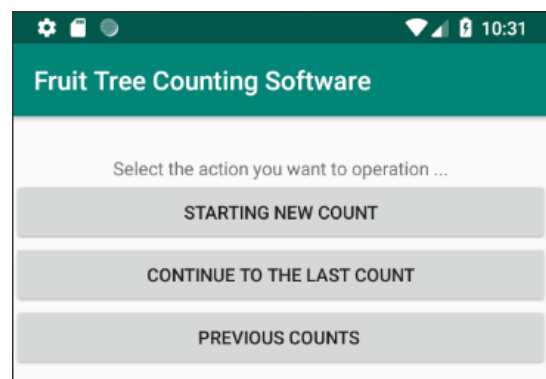


Fig.4. Application main menu

Since there may be more than one fruit tree and fruit species in a garden, when you enter from the "Starting

"New Count" menu, you are asked first how many trees will be counted in total (Fig. 5). After entering the number of trees, the tree name and type for each tree must be selected. The green button increases the count. The last number can be followed by the text component placed on the button. The red button is used to reduce the count. The counting is not terminated, assuming that the counting process will continue without pressing the "End" button. By pressing the "End" button, the count is terminated and the report is sent to the cloud system and listed under the Old count's menu.

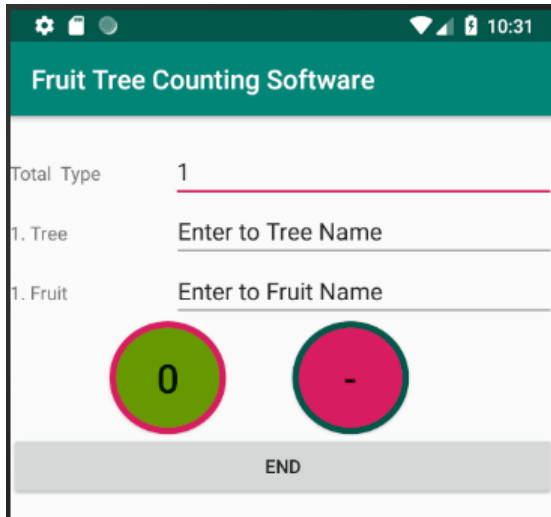


Fig.5. Starting new count menu

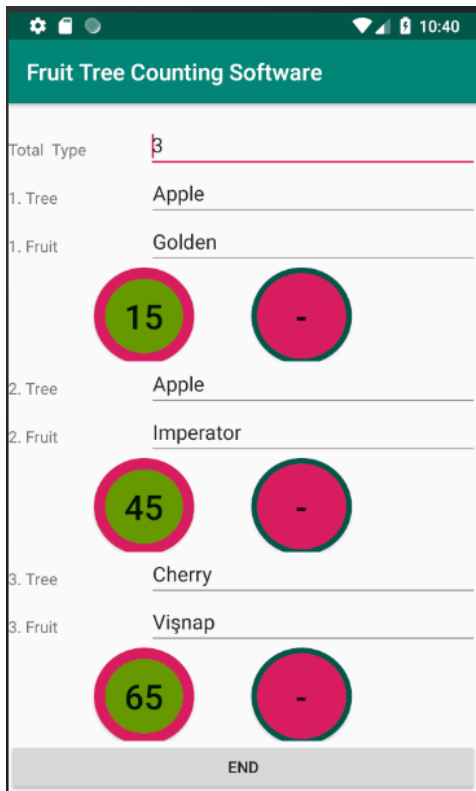


Fig.6. Continue to the last count menu

The "Continue to the last count" menu is the menu that will be active unless the last count is finished. After

entering this menu, the last active count information is displayed. Thus, the user can resume counting. As with the new count, it can increase with the green button and decrease with the red button. Finally, at the end of the count, the data can be saved to the cloud system by clicking on the "end" button (Fig. 6). It will be easier to count in long-distance with this feature.

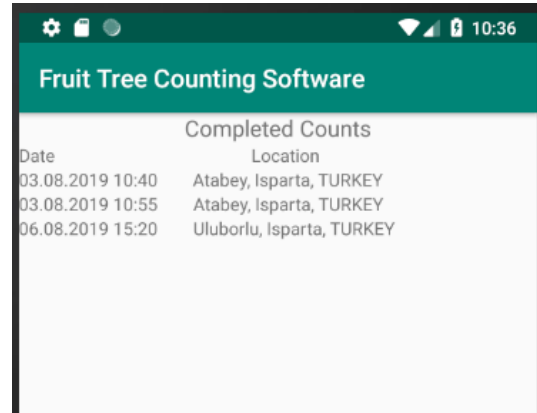
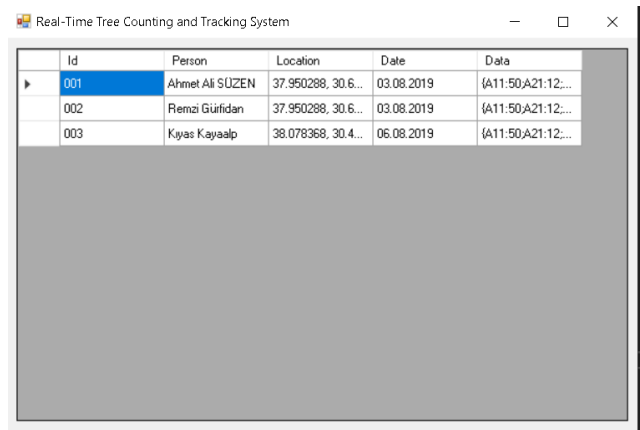
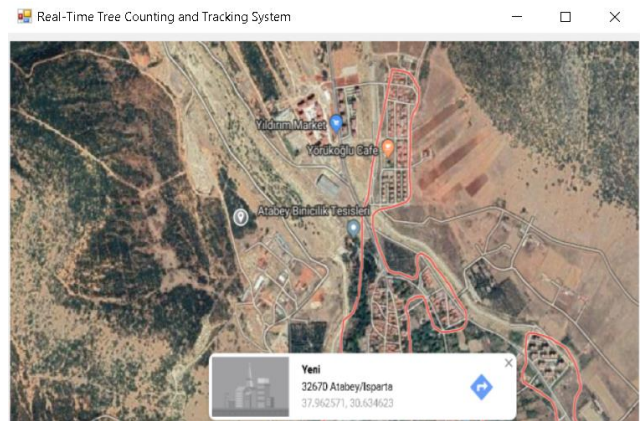


Fig.7. Previous counts menu



(a)



(b)

Fig.8. Desktop tracking software, (a) List of received data (b) Map display

The "Previous Counts" menu lists the completed counts. As shown in Fig. 7, the previous counts are sorted according to the dates. Locations that are included as addresses at the time of listing are pinned on Google Maps in the central application. In this way, the locations

where the counting is made can be visualized regionally.

While the mobile application is counting, the desktop application can show the data. It has been shown the interface of the application in Fig. 8. All data added to the cloud system can be tracked through the application and the coordinate data from GPS is marked via Google map.

The features of the tree counting applications developed as a result of this section are as follows.

- It can operate without an Internet connection. Updates when the Internet is available.
- Gets the GPS location at the start of each count.
- It uses SQL-Lite on the device and Firebase database in the cloud.
- It can count 50 different tree species at once.
- It has the ability to pause the count.
- It can see past counts from his phone or tablet.
- The count center can be monitored instantaneously on the computer.

## V. CONCLUSION

Trees are insured for risks from natural disaster and climate change. For this, the number of trees in the field should be known. Calculators are used for tree counting and counting applications for single tree species. But multiple tree species and long-distance areas are not the right solutions. In the literature studies, there are image processing based systems. In order to use them, the tree species must be taught in advance. Such applications have a high cost.

In this application study, cloud based Android application has been developed which will be used by the user for tree counting during open field study. Counts are performed for 50 different tree species at the same time. At the same time, the counting can be monitored simultaneously by the center. Thus, the problems of the counters during the counting are prevented and it is determined that the counters perform counting in the correct location. The use of a cloud-based and real-time firebase database is predicted to be an example of Big Data applications, except for solving the problems mentioned.

In order to determine the performance and accuracy of the application, the application was used by 100 people in Isparta. After a month of usage, 5 questions were directed to the users (Table 2). Questions are created from the problems that users experience with the subject. For the reliability of the survey, the results were interpreted in the SPSS. For reliability analysis, the Spearman-Brown formula given in Equation 1 was applied.

$$r_{SB} = \frac{2r_{xy}}{1+r} \quad (1)$$

Where  $r_{xy}$  is correlation between first half and second half.

We applied the test-retest method from the reliability analysis methods. If the correlation value found in this

method is positive and close to 1, it is decided that the test is reliable. The responses of the same test applied at different time intervals were subjected to The Related T test in the SPSS program and the correlation between the obtained observations was looked at. The results obtained in the SPSS software are given in Table 3. Since the correlation values given in the table are close to 1, it is believed that the test is reliable.

Table 2. Question and answer

	Yes	No
Online / Offline Mode Does it work?	100	0
Is the location correct?	98	2
Was the data transferred in real time?	98	2
Real-time location Is it shown on the map?	98	2
Did you receive an error from the application?	98	2

Table 3. SPSS Related T-test results for reliability

Paired Samples Correlations			
	N	Correlation	Sig
Pair 2	100	,812	,000
Pair 3	100	,704	,000
Pair 4	100	,704	,000
Pair 5	100	,812	,000

As a result of this assessment, users of the application have declared that it saves time and that confusion has greatly decreased in the count. In the next step of the study, we will provide feedback and performance evaluations for the data storage and reporting service that the application provides to its users. Improvements made through the feedback Also, land images will be captured to support the location. These images will be transmitted and monitored synchronously to the central server.

## REFERENCES

- [1] Çiftçi, T. "Tarım Sigortalarının Devlet Tarafından Desteklenmesi ve Tarım Sigortaları Havuzu Sistemi". *Ankara Barosu Dergisi*, (4), 2014.
- [2] Doğan, Z., Arslan, S., & Berkman, A. "Türkiye'de tarım sektörünün iktisadi gelişimi ve sorunları: tarihsel bir bakış". *Niğde Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 8(1), 29-41, 2015.
- [3] Ertan, A., & Mustafa, G. Ö. K. "Eğirdir İlçesi Tarım Üreticilerinin Tarım Sigortası Yaptırmaya Karar Verme Sürecinde Etkili Olan Faktörlerin Analizi". *ODÜ Sosyal Bilimler Araştırmaları Dergisi (ODÜSOBİAD)*, 3(5), 66-76, 2012.
- [4] Li, W., Fu, H., Yu, L., & Cracknell, A. "Deep learning based oil palm tree detection and counting for high-resolution remote sensing images." *Remote Sensing*, 9(1), 22, 2017.
- [5] Dorj, U. O., Lee, M., & Yun, S. S., "An yield estimation in citrus orchards via fruit detection and counting using image processing". *Computers and Electronics in Agriculture*, 140, 103-112, 2017.
- [6] Sotoca, J. M., Mollineda, R. A., & Sánchez, J. S. "A meta-learning framework for pattern classification by means of data complexity measures". *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, 10(29), 31-38, 2016.

- [7] Bresilla, K., Perulli, G. D., Boini, A., Morandi, B., Grappadelli, L. C., & Manfrini, L. "Single-shot convolution neural networks for real-time fruit detection within the tree". *Frontiers in plant science*, 10, 2019.
- [8] Fiona, J. R., & Anitha, J., "Automated Detection of Plant diseases and Crop Analysis in Agriculture using Image Processing Techniques: A Survey". In *2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)* (pp. 1-5). IEEE, February 2019.
- [9] Ünal, K., & Kurucu, Y., "Zeytin Ağaç Sayımında Yüksek Çözünürlüklü Uydu Görüntülerinin Kullanımı". *Zeytin Bilimi*, 2(1), 1-11, 2011.
- [10] Aydoğdu, M., Güzel, A., Sürücü, A., Aydoğdu, M. H., & Çullu, M. A., "Tektek Platoları Antepfistiği Florasının Ve Ağaç Sayılarının Uzaktan Algılama Ve Coğrafi Bilgi Sistemleri Teknikleri Kullanılarak Belirlenmesi". *Electronic Turkish Studies*, 8(12), 2013.
- [11] Santoro, F., Tarantino, E., Figorito, B., Gualano, S., & D'Onghia, A. M., "A tree counting algorithm for precision agriculture tasks". *International Journal of Digital Earth*, 6(1), 94-102, 2013.
- [12] Bazi, Y., Malek, S., Alajlan, N., & AlHichri, H., "An automatic approach for palm tree counting in UAV images". In *2014 IEEE Geoscience and Remote Sensing Symposium* (pp. 537-540). IEEE, 2014.
- [13] Li, W., Fu, H., Yu, L., & Cracknell, A., "Deep learning based oil palm tree detection and counting for high-resolution remote sensing images". *Remote Sensing*, 9(1), 22, 2017.
- [14] Wu, J., Ping, L., Ge, X., Wang, Y., & Fu, J., "Cloud storage as the infrastructure of cloud computing". In *2010 International Conference on Intelligent Computing and Cognitive Informatics* (pp. 380-383). IEEE, June, 2010.
- [15] Makhlouf, S. A., & Yagoubi, B., "Clustering Strategy for Scientific Workflow Applications in IaaS Cloud Environment". In *International Conference Europe Middle East & North Africa Information Systems and Technologies to Support Learning* (pp. 482-491). Springer, Cham, October, 2018.
- [16] Moroney, L., "The firebase real-time database". In *The Definitive Guide to Firebase* (pp. 51-71). Apress, Berkeley, CA, 2017.
- [17] Chatterjee, N., Chakraborty, S., Decosta, A., & Nath, A., "Real-time Communication Application Based on Android Using Google Firebase". *Int. J. Adv. Res. Comput. Sci. Manag. Stud.*, 2018.
- [18] Lahudkar, P., Sawale, S., Deshmane, V., & Bharambe, K., "NoSQL Database-Google's Firebase: A Review". *International Journal of Innovative Research in Science, Engineering and Technology*, 7(3), 2018.
- [19] Khedkar, S., Thube, S., Estate, W. I., & Naka, C., "Real time databases for applications". *International Research Journal of Engineering and Technology (IRJET)*, 4(06), 2078-2082, 2017.
- [20] Aksoy, B., "Three dimensional online virtual apparel internet page application". *Industria Textila*, 67(4), 256-259, 2017.

## Authors' Profiles



**Ahmet Ali SÜZEN** is head of cyber security application and researching center at Isparta University of Applied Sciences. He completed his doctorate in computer engineering. He has program development experience cyber security, deep learning and software development. He has 4 books in the field of deep learning and software development. He has been editor-in-chief of the journal in two 2 international journals in the field of engineering. He also has project management experience, having successfully completed 2 projects about cyber security and mobile software development.



**Remzi GÜRFİDAN** completed his bachelor's degree and master's degree in Electronic-Computer Education at Süleyman Demirel University. He interested in IoT and Mobil Programming. He is currently a Ph.D. student in the Computers Engineering at Süleyman Demirel University.



**Kıyas KAYAALP** completed his bachelor's degree in Electronic-Computer Education at Marmara University. He completed his master's degree in Electronic-Computer Education at Süleyman Demirel University. He completed his PhD in the Physics Department of Süleyman Demirel University. He has been working at the Uluborlu Selahattin Karasoy Vocational School since 2001. He has an international book titled "Deep Learning and his applications in Turkey". He also served in the Leonardo Vinci Mobility Program on computer security and ethical hacking.



**Mehmet Ali ŞİMŞEK** completed his bachelor's degree and master's degree in Computer Engineering at Süleyman Demirel University. He interested in IoT and deep learning. He is currently a Ph.D. student in the Computers Engineering at Istanbul University.

**How to cite this paper:** Ahmet Ali Süz en, Remzi Gürfidan, Kıyas Kayaalp, Mehmet Ali Şimşek, "Real-Time Tree Counting Android Application and Central Monitoring System", *International Journal of Information Technology and Computer Science(IJITCS)*, Vol.12, No.2, pp.18-23, 2020. DOI: 10.5815/ijitcs.2020.02.02