Smart home security system using IoT for resilient remote tracing and mitigation of thefts

Aditya Abeysinghe
Independent Researcher
abeysingheaditya4@gmail.com

***Abstract*—Today people increasingly rely on technology as a fast and cost-effective means of improving their standard of living. Remote home-theft protection is no longer a naive concept, given how readily current microcontroller-based systems can be operated through Android control. Existing systems, however, still lack several features expected of a truly "smart" home theft-security system, including real-time notifications, face identification, reliability of the underlying technologies and high usability. The main objective of this research is therefore to identify, analyze and implement a solution to these drawbacks. The proposed system uses a Passive Infrared (PIR) sensor, a two-way relay switch, an Arduino Uno board and an HC-05 Bluetooth module, which link the home and the homeowner through a real-time Firebase online database; an accompanying Android application controls the system remotely and serves as the platform on which users are notified when a theft occurs. Face detection is performed on the phone using Google's Vision API. An end-to-end functional test confirmed that the proposed system fixes issues of current systems: intrusion events were correctly recorded ensuring resilience and that the corresponding notifications are delivered to registered users in near real-time.**

***Index Terms*—Arduino, Bluetooth, face identification, Firebase, Internet of Things (IoT), real-time notifications, theft detection, smart home.**

I. INTRODUCTION

The proliferation of the Internet of Things (IoT) has transformed the ordinary residence into a connected environment in which sensing, actuation and remote monitoring can be coordinated from a single handheld device [1]. Home security is one of the most vital applications of this trend: burglary remains among the most frequently reported property crime, and the ability to detect an intrusion and alert the occupant instantly is of direct value to homeowners [1], [2]. Advances in low-cost microcontrollers, wireless communication modules and cloud back-ends have made it economically feasible to build such systems [2], [3].

A genuinely "smart" theft-detection system is today defined by the qualities end users expect of it—near real-time notifications, face identification rather than a bare motion alarm, reliable and well-understood technologies, and high resilience and availability. Yet, many existing microcontroller-based systems satisfy only a subset of these expectations, which motivates the present work [1], [8].

The first such weakness concerns the notification channel. A large proportion of reported designs rely on the Global System for Mobile Communications (GSM) to place a call or dispatch a Short Message Service (SMS) alert when a sensor is triggered [3], [4]. While such SMS/GSM alerts work, they are slow to arrive, cost money to send each time, and can carry only a short line of plain text, so the owner is told that the sensor was triggered but not what set it off or who was there. [4]. The second weakness is the absence of identity verification: systems built around Passive Infrared (PIR) motion sensing alone cannot distinguish a returning family member from an intruder, which produces frequent false alarms and erodes user trust [5], [6]. Only recently have designs begun to embed face identification to attach an identity to a detected event [6], [7]. Therefore, a theft-detection system that is unreachable at the moment of a break-in and is unable to notify in near real-time fails the end user's expectations.

The main objective of this research is therefore to identify, analyze and implement a solution to the aforementioned drawbacks of current systems. The proposed system is designed to be highly usable, economical, reliable, secure and easily configurable, so that it complies with current end-user expectations of a smart theft-detection system.

To meet these goals, the system combines inexpensive and widely available components with a managed cloud back-end, whose overall architecture is shown in Fig. 1. Table I summarizes how the proposed system addresses the three expectations identified above—the notification channel, identity verification, and reliability, resilience and availability—together with the technique used for each and the novelty it contributes.

TABLE I. Design Expectations Addressed by the Proposed System

Expectation addressed	Technique and its role in the system	Novelty / contribution
Notification channel	A Firebase real-time database links the Android application, pushing an alert the instant an event is detected.	Community-level alerting: real-time notifications reach the owner or family members, adding a social redundancy layer if the owner is unreachable, in contrast to single-recipient SMS/GSM designs.
Identity verification	The Android application activates the camera and performs face detection on the phone to establish who triggered a PIR event before an alert is raised.	On-device (edge) face identification: camera images are processed on the home-end phone and never leave the home, giving a privacy and cost advantage over cloud-vision or CCTV/GPU-based systems while suppressing false alarms.
Reliability, resilience and availability	The PIR sensors and Arduino Uno detect an intrusion and raise a theft alert; a managed, replicated Firebase back-end relays the alert to the owner and returns the owner's arming command to the Arduino, which drives the two-way relay to engage the magnet lock.	High availability from a managed, replicated cloud back-end rather than a self-hosted server that forms a single point of failure, built on low-cost, well-established components; on receiving the alert the owner can remotely arm the premises, giving a physical response beyond mere notification [10].

Building on the analysis of these feature gaps in current smartphone-linked home-theft systems, this paper makes two main contributions:

- the design of a low-cost architecture whose distinguishing features are: on-device (edge) face identification that keeps camera images within the home, a remote response in which the user, on receiving a theft alert, can arm the premises (a magnet lock) directly from the application rather than the system stopping at a notification, and community-level notification to the owner or family members
- a working implementation and evaluation of this design using an Arduino Uno, PIR sensor, a two-way relay switch, an HC-05 Bluetooth module and an Android/Firebase software stack

The remainder of this paper is organized as follows. Section II reviews related work and positions the proposed system against it. Section III presents the methodology, comprising the system architecture, the hardware and software components and the operational workflow. Section IV reports the implementation, Section V presents the results and discussion, Section VI concludes, and Section VII outlines directions for future work.

II. LITERATURE REVIEW

Prior work on home-theft protection can be grouped into four broad themes: microcontroller-based motion alarms, cloud-mediated real-time systems, vision-based identity verification, and studies of security, reliability and availability. This section surveys each theme and identifies the gap that the present work addresses.

A. Microcontroller and PIR-Based Alarm Systems

Liyakat et al. [5] described a PIR-sensor-based Arduino system that detects movement in a monitored zone and triggers a response accordingly; such designs are attractive for their simplicity and negligible cost, but they stop at raising an alarm and do not identify the source of the motion. Nayak and Dass [4] extended the idea with a GSM module, PIR, smoke and gas sensors and a password-protected lock, giving multi-hazard coverage and remote SMS alerting. Abiodun and Okpe [3] similarly pair an Arduino Uno with an ultrasonic sensor and a GSM/SIM800 module, sending an SMS to the owner upon detection and remaining silent otherwise. These works confirm the practicality of embedded sensing, yet the authors of [3] themselves note that the notification path could be improved for faster and easier alerting—underscoring the latency and low information-richness of SMS-centric designs.

B. Cloud-Mediated Real-Time Systems

Arnob, Latif et al. [10] presented a real-time, closed-loop IoT home-surveillance system for Android that uses Firebase as the intermediary between the sensing hardware and the mobile client, enabling near-instant updates and remote control. Such cloud-mediated systems improve responsiveness and create a live channel for the user rather than a one-way SMS, and they benefit by the availability of managed database services. However, many such prototypes still treat the cloud purely as a message relay and do not incorporate any means of identifying the person who caused an event.

C. Vision-Based Identity Verification

Rahim et al. [6] proposed Logit-Boosted convolutional-neural-network models that jointly perform anomaly detection and face recognition on smart-home IoT data, reporting strong accuracy for both tasks and demonstrating that an identity can be attached to a detected event. Sabit [7] described an Artificial Intelligence-based smart security system that combines PIR sensing, OpenCV-based video processing and facial recognition with automated user notifications while attempting to preserve occupant privacy. These systems show the value of face identification for suppressing false alarms, but they frequently depend on more capable hardware, such as single-board computers or GPUs, and do not always address cost, configurability or the resilience of the overall service.

D. Security, Reliability, Resilience and Availability

Touqeer et al. [8] analyzed security challenges, issues and solutions across the perception, network and application layers of smart-home IoT and highlighted the prevalence of single points of failure and weak communication security. Vardakis et al. [1] reviewed smart-home security using IoT and catalogue threats such as unauthorized access, data breaches and device tampering, together with encryption- and authentication-based countermeasures. To improve dependability, Soni and Singh [9] proposed a hybrid blockchain and smart-contract framework aimed specifically at resilient IoT security in smart homes, while the tertiary study by Pilotti et al. [2] synthesized recent secondary studies on Artificial Intelligence of Things-based security systems for smart homes and buildings and confirms that reliability, availability and intelligent detection are recurring open concerns.

E. Research Gap and Positioning

Table II summarizes how representative prior systems address the four expectations set out in Section I. The comparison reveals that no single low-cost, microcontroller-based design simultaneously delivers real-time notifications, face identification, and cloud-backed reliability and availability while remaining economical and easily configurable. The system proposed in this paper is positioned to fill this gap by combining an Arduino Uno / PIR / two-way-relay sensing layer, an HC-05 Bluetooth link and an Android front end with a Firebase real-time back-end, thereby uniting real-time notifications, face identification and high availability in a single economical and configurable package.

TABLE II. Feature Comparison of Representative Home-Theft Systems

System	Real-time notif.	Face ID	Reliability	Resil. / avail.	Cost
GSM/PIR Arduino [3]–[5]	No (SMS)	No	Yes	No	Low
Cloud / Firebase [10]	Yes	No	Med.	Med.	Low
Vision / AI [6], [7]	Yes	Yes	Med.	No	High
Proposed system	Yes	Yes	Yes	Yes	Low

III. METHODOLOGY

A. System Overview

The proposed system follows a modular, layered methodology in which a low-cost embedded sensing layer, a smartphone-based processing layer and a managed cloud layer cooperate to detect, identify and respond to intrusions. Fig. 1 shows the overall architecture together with the flow of data and commands between the components. Intrusion events originate at the Arduino-based home end, which are enriched with face-identification results by the home-end Android application. They are propagated in real-time through the Firebase database, and are delivered to the home owner. The design deliberately separates local sensing and actuation from cloud-based communication so that each layer can be developed and tested independently.

B. Hardware Components

The hardware layer is built entirely from inexpensive, widely available components. Their names and roles are summarized in Table III.

TABLE III. Hardware Components of the Proposed System

Component	Usage
Arduino Uno	Embedded controller; reads the PIR sensors and drives the actuation logic.
PIR motion sensor	Detects movement in the monitored area and triggers the detection logic.
Two-way relay switch	Engages or releases the magnet lock that secures the premises on the user's command.
HC-05 Bluetooth module	Wireless serial link between the Arduino and the home-end Android phone.
Android phone (home end)	Captures the image for face detection and runs the home-end application.

C. Software Components

The software layer comprises the embedded firmware, an on-device vision library, a cloud back-end and the mobile application, as listed in Table IV. Face detection at the home end is performed by the Android application with the Google's Mobile Vision API, so that captured images are processed locally and never leave the device.

TABLE IV. Software Components of the Proposed System

Component	Usage
Arduino IDE	Development environment used to write and upload the C firmware to the Arduino.
C language	Language of the Arduino firmware: sensor reading, relay actuation and Bluetooth I/O.
Google Mobile Vision API	Performs on-device face detection on the captured image; images are processed on the phone and never leave the device.
Firebase (real-time database and web services)	Cloud back-end for real-time data exchange, storage and push notifications.
Android application	The same application runs at both the home end and the home-owner end: it activates the camera and runs face detection, exchanges data with Firebase and forwards commands to the Arduino, and receives notifications, issues the arming command and the user may notifies neighbors and the police.

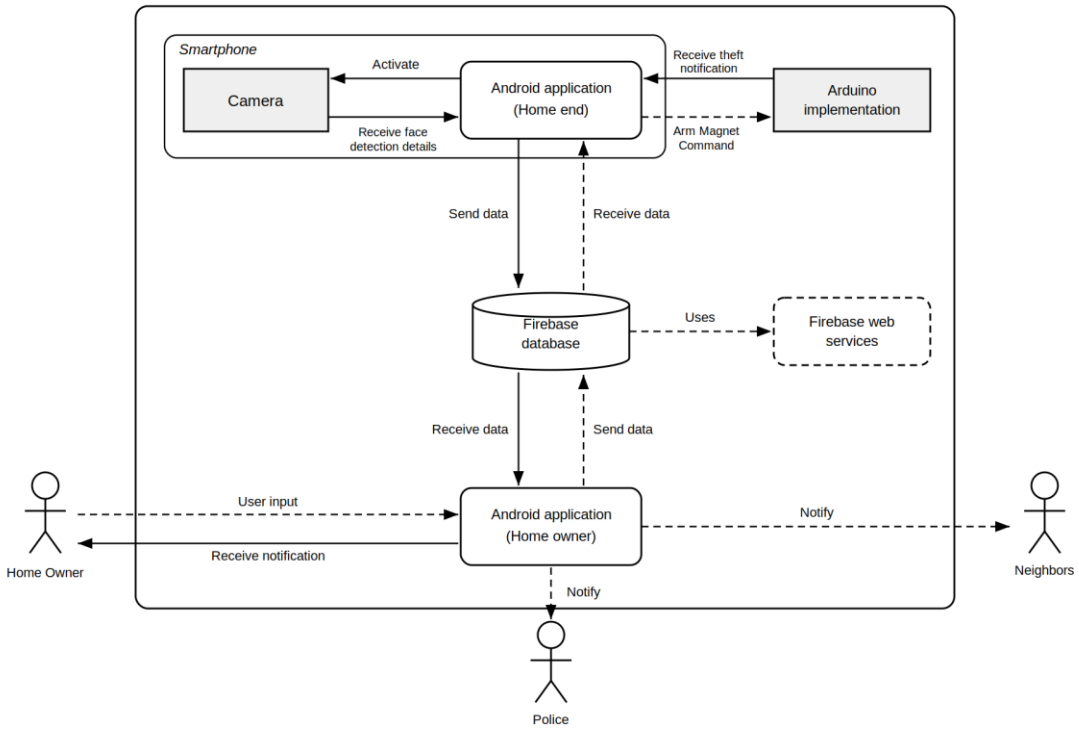

Fig. 1. System architecture of the proposed smart home theft-detection system.

D. Operational Workflow

The system operates as a detect–identify–notify–respond cycle. First, in the detection stage, a PIR sensor registering motion causes the Arduino to raise a theft event and transmit it over Bluetooth to the home-end Android application. Second, in the identification stage, the home-end application activates the camera and performs on-device face detection to determine whether it is a human; images are processed locally and are not uploaded. Third, in the notification stage, the event details—including the identification result—are written to the Firebase real-time database, which immediately notifies the application, providing the real-time, community-level alerting introduced in Section I. Finally, in the response stage, the owner reviews the alert and issues an arming command from the application; the command travels back through Firebase to the home-end application and over Bluetooth to the Arduino, which drives the two-way relay to trigger the magnet lock. This human response ensures that a physical countermeasure is taken only when the owner confirms a genuine threat, reducing the risk of acting on a false alarm.

E. System Algorithm

The end-to-end operation of the system is summarized in Algorithm 1.

Algorithm 1 Operation of the proposed smart home theft-detection system

```
Start
Turn on the Arduino system
Open the theft-detection Android application
repeat
    Log in to the system
    if login is invalid then
        Display invalid-login error message
    end if
until login is valid
Load the menu screen
Load home menu screen on home-based phone
loop
    Wait for intrusion detection
    if intrusion detected then
        Activate camera and wait for face detection
        if face detected then
            Add entry to the Firebase database
            Receive notification on app
            Control system (owner selects a response):
            if arm magnet then
                Send arming command to the database
                Arduino receives the command
                Arm / disarm magnet lock via two-way relay
            else if notify police then
                Load the phone dialer
            else if notify neighbours then
                Send an SMS to all neighbours
            end if
            if logout then exit loop
        end if
    end if
end loop
End
```

IV. IMPLEMENTATION

A. Hardware Assembly

The hardware was assembled around a single Arduino Uno board, as shown in Fig. 2. Every module was wired directly to the Arduino except for its power and ground lines, which were routed through a breadboard: all ground pins were tied to the breadboard's negative rail, and all supply pins to the positive rail. The PIR sensor output was read on the digital pin 3 of the board.

B. Detection and Notification

In the prototype, the PIR sensor becomes active after a 10-second calibration period. A detected motion drives the PIR pin high, and returns it low when the motion ends; the Arduino firmware relays this state to the home-end phone over the HC-05 Bluetooth module. On receiving an intrusion signal, the phone activates the camera and runs face detection through Google's Vision API, and a positive detection adds an entry to the online Firebase database. A notification is raised on the owner's phone and on the phones of family members authenticated to the same account, so that the household can respond even if the owner misses the alert.

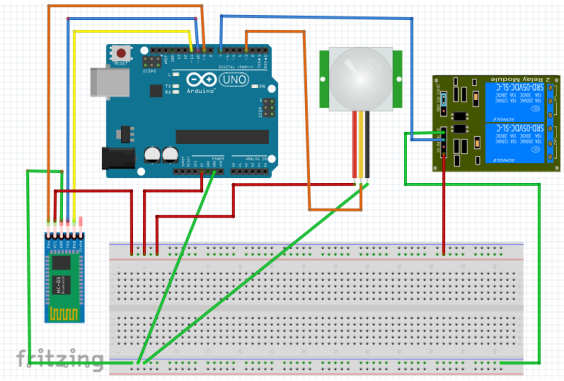

Fig. 2. Arduino connections with the other hardware (Fritzing diagram).

C. Remote Control and Actuation

Through the Android application, the owner can arm or disarm the magnet, notify the police or selected neighbors, and update the status of a database entry once a theft is resolved so that further notifications cease. Arming or disarming requires the login password to be re-entered, so that a hacker in possession of an unlocked, logged-in phone still cannot actuate the lock. When an arm/disarm command is issued, the corresponding database entry is updated, the home-end phone receives the change and forwards it to the Arduino over the HC-05 module, and the relay switch locks or unlocks the magnet.

Fig. 3 shows the assembled prototype. The two-way relay switch is connected to a 3V external supply (2 × 1.5V AA batteries for demonstration) and to the electromagnet. When the remote command reaches the Arduino, it drives the relay: activating the relay energizes the electromagnet from the external supply and locks the door, while deactivating it terminates the current and releases the lock. Powering the magnet from a separate supply keeps its high-current path isolated from the microcontroller.

D. Security

Security was enforced at several levels. Authentication is restricted to username-and-password login—social-media logins were deliberately removed so that a compromised phone cannot be used to sign in—passwords are stored using hashing, and all data exchange is protected through Firebase's secure communication. As a final safeguard, the arm/disarm action itself is gated by a second password entry, so that physical control of the lock is never available from an unlocked phone alone.

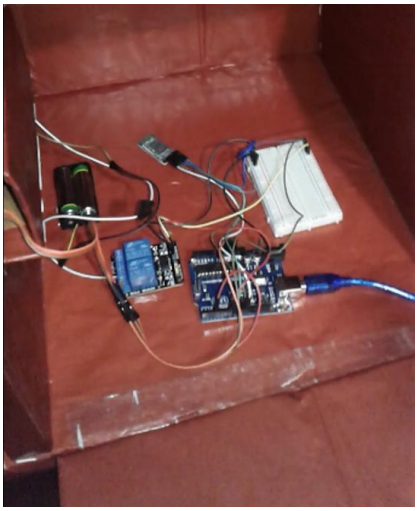

Fig. 3. Assembled prototype.

E. Source Code Availability

The complete source code of both the Android application and the Arduino firmware is maintained in a public repository and is available at https://github.com/aditya1962/RealTimeHomeTheftDetect.

V. RESULTS AND DISCUSSION

The integrated prototype was evaluated through functional (black-box) testing, in which each stage of the detect–identify–notify–respond pipeline was exercised and the observed behavior compared against the expected outcome. A representative end-to-end test placed a subject in the monitored area, first traversing the PIR sensor's detection field and then presenting to the home-end camera (Fig. 3), so that motion sensing, face detection, database update and notification were triggered in sequence. The detection, notification, actuation and authentication functions were then tested individually. Table V summarizes the ten test cases and their outcomes.

All ten cases produced the expected behavior. These results verify that the hardware and software components interoperate correctly and that the system realizes its intended behavior. The evaluation was functional rather than quantitative; measurement of detection-to-notification latency and face-detection accuracy over a larger trial is left to future work.

TABLE V. Functional Test Cases and Outcomes

Test case	Action	Expected outcome	Result
TC1 End-to-end intrusion	Subject crosses the PIR field, then presents to the camera	Face detected and the event recorded in Firebase	Pass
TC2 Motion without a face	Subject crosses the PIR field but no detectable face is presented	No database entry created	Pass
TC3 Idle scene	No movement in the monitored area	PIR stays inactive; no event logged	Pass
TC4 Owner notification	An intrusion event is recorded	Push notification received on the owner's phone	Pass
TC5 Multi-device sync	Event recorded / status changed on one device	Family member's phone receives the same alert; change reflected on both	Pass
TC6 Arm magnet	Owner selects Arm and enters the correct password	Relay energizes; electromagnet locks the door	Pass
TC7 Disarm magnet	Owner selects Disarm with the correct password	Relay de-energizes; door unlocks	Pass
TC8 Update status	Owner marks the event as resolved	No further notifications for that event	Pass
TC9 Login authentication	Log in with correct, then incorrect, credentials	Access granted for valid; error shown for invalid	Pass
TC10 Password-gated actuation	Attempt arm/disarm with an incorrect password	Action rejected; magnet state unchanged	Pass

VI. CONCLUSION

This research explored the potential of combining on-device face detection, an online real-time database and real-time user notifications to build a modern remote home theft-detection system. Selecting Bluetooth for the local link between the home and the controller—rather than GSM or Wi-Fi—keeps that link free of any external-provider dependency and contributes to the reliability and ease of use of the system. A working prototype was implemented and shown to be secure, highly usable and easily configurable; built on Android as the mobile platform, the solution is well positioned to reach a large target market.

VII. FUTURE WORK

Several enhancements are foreseen for the proposed system. First, the mobile application could be extended beyond Android to the iOS and Windows platforms, so that a larger user base can benefit from its capabilities. Second, additional functionality could be incorporated, including online video sharing through a live-streaming or hosting service, giving the owner a real-time visual feed of a detected event. Third, an alternative yet secure authentication mechanism, such as fingerprint scanning, could be introduced to complement or replace password-based entry.

Further work should also strengthen the resilience of the system, which at present relies on a single Arduino board, a single home-end phone and one HC-05 Bluetooth link—each a potential single point of failure—and on a cloud connection for remote notification. Further, store-and-forward buffering on the home-end phone would retain events locally during a connectivity loss and synchronize them once the link is restored, so that no event is lost. Furthermore, a redundant Wi-Fi or GSM path could serve as a fallback for the remote-notification tier when the primary channel is unavailable. Together, these measures would move the system from a resilient prototype towards a highly available deployment.

REFERENCES

[1] G. Vardakis, G. Hatzivasilis, E. Koutsaki, and N. Papadakis, "Review of Smart-Home Security Using the Internet of Things," Electronics, vol. 13, no. 16, Art. no. 3343, 2024.

[2] F. Pilotti, A. Pavone, L. Di Sabatino Farinelli, S. Tinelli, and G. Paolone, "AIoT-Based Security Systems for Smart Homes and Smart Buildings: A Tertiary Study," Future Internet, vol. 18, no. 6, Art. no. 307, 2026.

[3] O. J. Abiodun and A. O. Okpe, "Smart Home Security using Arduino-based Internet of Things (IoTs) Intrusion Detection System," World Journal of Advanced Research and Reviews, vol. 22, no. 3, pp. 857–864, 2024.

[4] M. Nayak and A. K. Dass, "GSM and Arduino based Smart Home Safety and Security System," Recent Trends in Information Technology and its Application, vol. 6, no. 1, pp. 1–6, 2023.

[5] K. S. S. Liyakat, K. K. S. Liyakat, G. M. Kosgiker, and V. D. Gund, "PIR Sensor-Based Arduino Home Security System," Journal of Instrumentation and Innovation Sciences, vol. 8, no. 3, pp. 33–37, 2023.

[6] A. Rahim, Y. Zhong, T. Ahmad, S. Ahmad, P. Pławiak, and M. Hammad, "Enhancing Smart Home Security: Anomaly Detection and Face Recognition in Smart Home IoT Devices Using Logit-Boosted CNN Models," Sensors, vol. 23, no. 15, Art. no. 6979, 2023.

[7] H. Sabit, "Artificial Intelligence-Based Smart Security System Using Internet of Things for Smart Home Applications," Electronics, vol. 14, no. 3, Art. no. 608, 2025.

[8] H. Touqeer, S. Zaman, R. Amin, M. Hussain, F. Al-Turjman, and M. Bilal, "Smart Home Security: Challenges, Issues and Solutions at Different IoT Layers," The Journal of Supercomputing, vol. 77, no. 12, pp. 14053–14089, 2021.

[9] S. Soni and A. Singh, "A Hybrid Blockchain and Smart Contract Framework for Resilient IoT Security in Smart Homes," Frontiers in Blockchain, vol. 8, Art. no. 1707911, 2025.

[10] S. S. Arnob, A. B. Latif et al., "A Real-Time Controlled Closed Loop IoT Based Home Surveillance System for Android using Firebase," in Proc. IEEE Int. Conf. on Robotics, Automation, Artificial-intelligence and Internet-of-Things (RAAICON), Dhaka, Bangladesh, 2019.