IAES Journal of Electrical and Electronics Engineering
Vol. 6, No. 1, June 2019, pp. 18~22
ISSN: 2338-7432, DOI: 10.11591/jeee.v6i1.pp18-22

Android-based IP camera monitoring systems: architecture, video streaming, remote access, and security challenges

Hendril Satrian Purnama[1], Tri Wahono[1], Wadipta Lingga Purusua[2]
[1]Embedded Systems and Power Electronics Research Group, Yogyakarta, Indonesia
[2]Faculty of Industrial Technology, Universitas Ahmad Dahlan, Yogyakarta, Indonesia

Article Info

Article history:

Received Jan 24, 2018
Revised Jul 10, 2018
Accepted Aug 31, 2018

Keywords:

Android
Cybersecurity
IP camera
Remote monitoring
Video streaming
Video surveillance
Wireless networks

ABSTRACT

Android-based IP camera systems combine network cameras with smartphone applications to provide remote video monitoring, camera control, and event notification. Their appeal lies in the availability of low-cost IP cameras, widespread Android devices, and wireless broadband connectivity. Nevertheless, practical deployment involves challenges that extend beyond displaying a live video stream. Application responsiveness depends on camera interoperability, streaming protocols, network bandwidth, latency, packet loss, power consumption, and the security of remote access. This review examines the architecture and development of Android-based IP camera applications, covering camera connectivity, video transport, user-interface functions, event-driven monitoring, system testing, and cybersecurity. Literature published through 2018 is considered, including studies of Android monitoring applications, IP-camera surveillance, wireless video systems, mobile multimedia delivery, and camera privacy. The evidence indicates that Android applications can improve the accessibility and flexibility of surveillance, particularly for homes, buildings, and traffic observation. However, unstable connectivity, inconsistent camera interfaces, excessive data use, and weak authentication can undermine reliability and user trust. Future implementations should prioritize interoperable protocols, adaptive streaming, secure communication, efficient notifications, and systematic evaluation under realistic network conditions and real deployments.

This is an open access article under the CC BY-SA license.

1. INTRODUCTION

The development of network-connected cameras has changed how individuals and organizations monitor buildings, residential environments, and public facilities. Unlike conventional analog CCTV systems, an Internet Protocol (IP) camera can transmit digital video through wired or wireless networks and can be accessed remotely using compatible client devices. Android smartphones provide a convenient monitoring interface because they combine network connectivity, video decoding, touchscreen interaction, and notification capabilities in one device. Consequently, Android-based IP camera systems have become relevant to building security, home monitoring, traffic observation, and other remote surveillance applications [1]–[5].

However, an effective monitoring application requires more than displaying a camera image on a smartphone screen. The system must establish reliable communication with the camera, receive and decode video streams, manage user access, and maintain acceptable performance under changing network conditions. Previous research on mobile IP-camera applications has demonstrated the feasibility of remote monitoring, but also highlighted the importance of stable Internet connections and appropriate communication mechanisms [1], [2]. This review therefore examines the architecture, communication technologies, application functions, performance considerations, and security requirements of Android-based IP camera systems, with the literature restricted to publications available up to 2018.

2. SYSTEM ARCHITECTURE AND OPERATING PRINCIPLES

2.1. General system architecture

An Android-based IP camera monitoring system generally consists of an IP camera, a network infrastructure, an Android application, and an optional recording or management server. The camera captures video, performs image encoding, and transmits the resulting data through an IP network. The Android device acts as a client that receives the stream, displays live video, and provides access to supported camera-control functions. Depending on the implementation, the system may also include a router, cloud service, local storage, motion detector, or gateway for managing multiple cameras [3]–[6].

The architecture can operate through a local area network or through remote Internet access. In a local configuration, the smartphone and camera communicate through the same Wi-Fi router, which generally simplifies device discovery and reduces dependence on external services. Remote monitoring requires a network path between the smartphone and camera, potentially involving a public IP address, dynamic DNS, a VPN, or a cloud-based relay service. Studies of wireless surveillance and cloud-connected cameras indicate that network topology and service configuration influence both system accessibility and security exposure [7]–[10].

2.2. Android application functions

The Android application acts as the main user interface for accessing and controlling the IP camera. Typical functions include camera registration, live video display, connection management, image capture, video recording, and access to stored footage. More advanced applications may provide pan-and-tilt control, motion notifications, camera grouping, and user authentication. Indonesian research on embedded building-security systems has demonstrated the integration of Android applications, IP-camera streaming, and remote control functions within a single monitoring platform [1]. The design of the application interface affects how efficiently users can respond to surveillance events. Camera lists, clear connection indicators, accessible playback controls, and understandable notification messages can reduce operational complexity. Systems monitoring multiple locations may also benefit from map-based camera identification and organized camera grouping. Widyantara *et al.* [5] demonstrated a video surveillance management application that integrated IP-camera information, live streaming, and geographical visualization, illustrating the value of combining video access with contextual location information.

3. VIDEO COMMUNICATION AND NETWORK TECHNOLOGIES

3.1. Streaming protocols

Video streaming is a central component of Android-based IP camera systems because it determines how camera data reaches the smartphone. Real-time streaming protocol (RTSP) is commonly associated with controlling streaming sessions, while real-time transport protocol (RTP) can carry the media packets. HTTP-based approaches can also support video delivery, particularly when streaming is integrated with web services or application programming interfaces. Early research on mobile IP-camera monitoring examined the use of HTTP and RTSP to make camera streams accessible through mobile devices [11].

The selection of a streaming method affects startup delay, buffering, packet-loss sensitivity, and compatibility with Android playback components. RTSP-based streaming can support low-latency live monitoring, but its operation may be affected by firewall restrictions and network address translation. HTTP-based delivery can be easier to integrate with conventional web infrastructure, although the streaming architecture and buffering strategy must be designed for real-time requirements. Research comparing RTSP, MPEG-DASH, and RTMP has shown that streaming protocols can produce different quality-of-experience outcomes under mobile network conditions [12].

3.2. Wireless connectivity and video quality

Wi-Fi and cellular networks provide the connectivity required for remote monitoring, but their performance can vary substantially. Signal strength, interference, available bandwidth, packet loss, and network congestion influence the continuity of live video. High-resolution streams require more bandwidth and processing resources, which may increase buffering and delay when network capacity is limited. Research on wireless multimedia transmission has identified quality of service and quality of experience as important considerations for mobile video applications [13], [14].

Adaptive video delivery can help maintain a usable monitoring experience when network capacity changes. A system may adjust video resolution, frame rate, or encoding bitrate according to the available bandwidth and playback-buffer condition. This approach can reduce interruptions, although aggressive bitrate reduction may make important visual details harder to identify. Studies of adaptive HTTP streaming and mobile video delivery provide useful design principles for Android surveillance applications operating over variable 4G and Wi-Fi connections [15]–[17].

4. APPLICATION AREAS AND SYSTEM INTEGRATION

4.1. Building and residential security

Residential and building monitoring are important application areas for Android-based IP camera systems. Users can inspect rooms, entrances, corridors, and outdoor areas without remaining physically present at the monitoring location. Integration with embedded controllers can extend the application beyond video display by enabling remote door control or coordination with other security devices. Asmara *et al.* [1] demonstrated an Android-based building-security application integrating IP-camera streaming, database services, and remote door-control functions.

Home automation provides further opportunities for combining cameras with sensors and communication services. Motion detectors can activate recording or trigger notifications when activity is detected in a monitored area. Gas sensors and other environmental devices can also be integrated into a broader security platform to provide information beyond visual monitoring. Supriyadi's Android-based home automation design illustrates how IP-camera streaming, motion detection, and alert communication can be combined within a Raspberry Pi-based system [3].

4.2. Traffic and public-space monitoring

Android-based IP-camera applications can also support traffic observation and monitoring of geographically distributed locations. Camera streams provide visual information that can help users assess traffic conditions, identify congestion, and observe events in selected areas. Distria *et al.* [2] developed an Android application for monitoring traffic density using IP cameras, demonstrating the relevance of mobile access to traffic surveillance infrastructure. A geographically organized monitoring interface can further simplify access to cameras distributed across multiple locations [5].

4.3. Event-driven monitoring

Continuous video recording can consume substantial storage capacity and network resources, particularly when several cameras operate simultaneously. Event-driven monitoring addresses this issue by activating recording or camera transmission when a relevant event is detected. Motion sensors, image-processing algorithms, or other contextual inputs can be used to determine when video capture is necessary. Xuan *et al.* [18] proposed an event-triggered smart-home video recording system using collaborative sensing and motion detection to reduce unnecessary recording.

Event-driven functions can be connected to Android notifications to improve the usefulness of remote monitoring. Instead of requiring users to watch a live stream continuously, the application can provide alerts when motion or another predefined event occurs. The camera may then transmit a live stream or recorded clip when the user requests further inspection. Such an approach can reduce unnecessary data consumption, although event detection must be evaluated carefully to avoid excessive false alarms or missed events [6], [18].

5. SECURITY, PRIVACY, AND RELIABILITY

Security is a fundamental requirement because an IP camera may expose private video and information about the activities of people in a monitored environment. Weak passwords, outdated firmware, insecure remote-access services, and poorly configured network interfaces can create opportunities for unauthorized access. Research on CCTV and IP-camera security has documented vulnerabilities involving embedded firmware, network communication, and device configuration [8], [9]. Studies of cloud-based IP cameras have also emphasized that remote connectivity introduces privacy and security concerns that must be addressed during system design [10].

Android applications must therefore implement secure authentication and responsible handling of camera credentials. Passwords should not be stored in plain text, and communication between the application, camera, and supporting server should use appropriate security protections. Users should be able to manage access permissions and remove devices that are no longer authorized. Research on smart-camera vulnerabilities and wireless-camera privacy has shown that security weaknesses can affect not only video confidentiality but also information about household occupancy and user behavior [19], [20].

Reliability also depends on the quality of the camera, network infrastructure, Android device, and application software. A monitoring application should handle disconnections, expired sessions, camera restarts, and changes in network availability without crashing or presenting misleading information. Testing should include different smartphone models, Android versions, network conditions, and camera

configurations. Previous Android monitoring research has demonstrated the importance of testing across multiple devices and maintaining a stable Internet connection for reliable operation [1].

6. RESEARCH CHALLENGES AND FUTURE DIRECTIONS

The literature indicates several challenges that remain important for Android-based IP camera development. Interoperability is a major concern because camera manufacturers may implement different streaming profiles, authentication methods, and control interfaces. Network variability can also reduce video quality, while continuous streaming may increase smartphone battery consumption and mobile-data usage. Research on mobile multimedia delivery has shown that video quality, energy consumption, and network-resource management must be considered together rather than as independent design problems [13], [16], [17].

Future systems should emphasize standardized interfaces, adaptive streaming, efficient event notification, and stronger security mechanisms. Camera discovery and configuration should be simplified so that users can connect compatible devices without complex manual procedures. Edge processing may reduce the amount of video transmitted by performing motion detection or preliminary event analysis near the camera. Wireless surveillance research has demonstrated the potential of distributing video processing and prioritizing network traffic to improve the efficiency of large-scale monitoring systems [7], [21].

7. CONCLUSION

Android-based IP camera systems provide a flexible approach to remote surveillance by combining network cameras, wireless communication, and smartphone applications. Their applications include residential security, building monitoring, traffic observation, and event-driven recording. Nevertheless, system effectiveness depends on interoperability, streaming performance, network reliability, application usability, and protection against unauthorized access. Future development should therefore combine adaptive video delivery, secure communication, efficient event processing, and systematic testing to produce monitoring systems that are reliable, practical, and suitable for real-world deployment.

REFERENCES

[1] S. B. Asmara, A. N. Jati, and U. A. Ahmad, "Design and implementation of a mobile application for monitoring and controlling an embedded-based building security system," *eProceedings of Engineering*, vol. 3, no. 1, 2016.

[2] E. Distria, R. Rumani, and C. Setianingsih, "Design of an Android-based mobile application for monitoring traffic density using IP cameras," *eProceedings of Engineering*, vol. 4, no. 3, 2017.

[3] T. Supriyadi, "Development of a Raspberry Pi-based home automation system with an Android smartphone user interface, integrated with GSM, WLAN, and internet communication networks," *Proc. Industrial Research Workshop and National Seminar*, 2018.

[4] Suryansyah, "Design and development of a WiFi-based environmental security monitoring system using IP cameras," *Journal of Electrical Engineering, Energy, and Information Technology*, vol. 3, no. 1, 2015.

[5] I. M. O. Widyantara, N. U. Wedanti, and I. B. A. Swamardika, "Design and implementation of a Web-GIS-based video surveillance system application," *Majalah Ilmiah Teknologi Elektro*, vol. 14, no. 1, 2015, doi: 10.24843/mite.2015.v14i01p09.

[6] H. Abbas Khalaf, A. S. Tolba, and M. Z. Rashid, "Event triggered intelligent video recording system using MS-SSIM for smart home security," *Ain Shams Engineering Journal*, vol. 9, no. 4, pp. 1527–1533, Dec. 2018, doi: 10.1016/j.asej.2016.10.001.

[7] T. Zhang, A. Chowdhery, P. (Victor) Bahl, K. Jamieson, and S. Banerjee, "The design and implementation of a wireless video surveillance system," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, 2015, pp. 426–438, doi: 10.1145/2789168.2790123.

[8] A. Costin, "Security of CCTV and video surveillance systems: threats, vulnerabilities, attacks, and mitigations," in *Proceedings of the 6th International Workshop on Trustworthy Embedded Devices*, Oct. 2016, pp. 45–54, doi: 10.1145/2995289.2995290.

[9] R. Alharbi and D. Aspinall, "An IoT analysis framework: an investigation of iot smart cameras' vulnerabilities," in *Living in the Internet of Things: Cybersecurity of the IoT - 2018*, 2018, pp. 47 (10 pp.)–47 (10 pp.), doi: 10.1049/cp.2018.0047.

[10] J. Liranzo and T. Hayajneh, "Security and privacy issues affecting cloud-based IP camera," in *2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON)*, Oct. 2017, pp. 458–465, doi: 10.1109/uemcon.2017.8249043.

[11] S. M. Putra, Handoko, R. Mandasari, and B. P. Bestari, "Analysis and design of an IP camera monitoring application using the HTTP protocol on mobile phones," *Proc. Seminar Nasional Aplikasi Teknologi Informasi*, 2010.

[12] A. Aloman, A. I. Ispas, P. Ciotirnae, R. Sanchez-Iborra, and M. D. Cano, "Performance evaluation of video streaming using MPEG DASH, RTSP, and RTMP in mobile networks," in *2015 8th IFIP Wireless and Mobile Networking Conference (WMNC)*, Oct. 2015, pp. 144–151, doi: 10.1109/wmnc.2015.12.

[13] H. Chaari, K. Mnif, and L. Kamoun, "Multimedia quality transmission evaluation over wireless networks: a survey," *Wireless Networks*, vol. 22, no. 8, pp. 2607–2621, Nov. 2015, doi: 10.1007/s11276-015-1127-5.

[14] H. Luo and M.-L. Shyu, "Quality of service provision in mobile multimedia - a survey," *Human-centric Computing and Information Sciences*, vol. 1, no. 1, Nov. 2011, doi: 10.1186/2192-1962-1-5.

[15] L. Arun Raj, D. Kumar, H. Iswarya, S. Aparna, and A. Srinivasan, "Adaptive video streaming over HTTP through 4G wireless networks based on buffer analysis," *EURASIP Journal on Image and Video Processing*, vol. 2017, no. 1, 2017, doi: 10.1186/s13640-017-0191-4.

[16] M. Seyedebrahimi, X.-H. Peng, and R. Harrison, "A quality driven framework for adaptive video streaming in mobile wireless networks," in *2014 IEEE Wireless Communications and Networking Conference (WCNC)*, Apr. 2014, pp. 2994–2999, doi: 10.1109/wcnc.2014.6952956.
[17] M. A. Hoque, M. Siekkinen, J. K. Nurminen, M. Aalto, and S. Tarkoma, "Mobile multimedia streaming techniques: QoE and energy saving perspective," *Pervasive and Mobile Computing*, vol. 16, pp. 96–114, Jan. 2015, doi: 10.1016/j.pmcj.2014.05.004.
[18] S. Xuan, D.-H. Park, and D. Kim, "A service platform for real-time video streaming based on RESTful API in IoT environments," *International Journal of Control and Automation*, vol. 10, no. 3, pp. 181–192, Mar. 2017, doi: 10.14257/ijca.2017.10.3.15.
[19] X. Ji, Y. Cheng, W. Xu, and X. Zhou, "User presence inference via encrypted traffic of wireless camera in smart homes," *Security and Communication Networks*, vol. 2018, pp. 1–10, 2018, doi: 10.1155/2018/3980371.
[20] A. Tekeoglu and A. S. Tosun, "Investigating security and privacy of a cloud-based wireless IP camera: NetCam," Aug. 2015, doi: 10.1109/icccn.2015.7288421.
[21] N. Abbas and F. Yu, "Design and implementation of a video surveillance system for linear wireless multimedia sensor networks," in *2018 IEEE 3rd International Conference on Image, Vision and Computing (ICIVC)*, 2018, pp. 524–527, doi: 10.1109/icivc.2018.8492776.