Journal of Modern Technology and Engineering
Vol.10, No.2, 2025, pp.102-110
https://doi.org/10.62476/jmte.10202

ENHANCING TRANSPARENCY IN BATNA'S PUBLIC TRANSPORTATION: A COST-EFFECTIVE REAL-TIME BUS TRACKING AND ROUTE OPTIMIZATION SYSTEM

Zineb Nedjla Alloui*, **Fetima Bourebbou, Larbi Guezouli**

LEREESI Laboratory, HNS-RE2SD, Batna, Algeria

Abstract: Public transportation in Batna suffers from a lack of transparency, leading to inefficiencies, unreliable schedules, and a poor commute experience. This thesis presents a comprehensive and cost-effective bus tracking system that aims to modernize public transportation in the city. The proposed solution integrates a GPS-enabled Raspberry Pi device for real-time location tracking, an AI-powered backend that uses long-short-term memory (LSTM) networks for accurate estimated arrival time (ETA) predictions, and a mobile Android application for user interaction. The system also incorporates Dijkstra's algorithm to offer optimized route planning. The emphasis was placed on low-cost hardware, lightweight web frameworks such as Flask, and secure communication via HTTPS and DDNS. The developed platform was tested and validated through real-world simulations, demonstrating significant improvements in route accuracy, ETA reliability. This work offers a modular and scalable foundation for improving public transportation in resource-constrained urban environments.

Keywords: Public Transportation, Bus Tracking, Artificial Intelligence, LSTM, Route Optimization, Smart Cities.

AMS Subject Classification: 68T01, 90B06, 90B20.

Corresponding author: Z.N. Alloui, LEREESI Laboratory, HNS-RE2SD, Batna, Algeria,
e-mail: *alloui.zineb.nedjla@hns-re2sd.dz*

Received: 21 June 2025; Revised: 12 July 2025; Accepted: 22 July 2025; Published: 11 September 2025.

1 Introduction

Public transportation plays a pivotal role in ensuring equitable urban mobility, yet in mid-sized cities such as Batna, Algeria, systemic inefficiencies—ranging from ambiguous routes and inconsistent schedules to the absence of real-time information—continue to impede accessibility and undermine user confidence. With 13 active bus lines serving over 5,000 daily commuters, the majority of whom are students and low-income residents, these shortcomings contribute to frequent delays and growing public dissatisfaction.

This study introduces a cost-effective and scalable bus tracking and route optimization system designed to enhance transparency and reliability in urban transit networks. The proposed solution integrates real-time GPS tracking, machine learning-based Estimated Time of Arrival (ETA) prediction using Long Short-Term Memory (LSTM) networks, and graph-based shortest-path routing via Dijkstra's algorithm.

Built with low-cost embedded hardware and open-source software components, the system is designed to offer high performance and reliability while remaining accessible for cities with limited technological infrastructure. Its modular design ensures ease of deployment and scalability across varying urban transport contexts.

To address data scarcity, a custom-developed Bus Route Collector mobile application was implemented to gather GPS traces from active routes. Security is ensured through the use of HTTPS protocols and Firebase Authentication. Experimental validation demonstrates the system's effectiveness, with 99.18% ETA prediction accuracy and sub-3.1 ms routing performance, surpassing existing benchmarks (Chughtai et al., 2022; As & Mine, 2018).

The remainder of this paper is structured as follows: Section 2 reviews relevant literature and existing technologies; Section 3 outlines the proposed system architecture and data methodology; Section 4 presents the experimental evaluation; Section 5 discusses findings, limitations, and deployment considerations; and Section 6 concludes with directions for future work.

2 Related Work

Public transportation systems worldwide increasingly leverage advanced technologies to improve reliability and user experience. However, many solutions remain too costly or resource-intensive for mid-sized, budget-constrained cities like Batna. This section critically reviews existing approaches in bus tracking, estimated time of arrival (ETA) prediction, which refers to forecasting the time a vehicle is expected to reach a specific stop based on current and historical data, route optimization, mapping, and security, highlighting their limitations.

2.1 Public Transportation Systems

Sophisticated systems in Singapore and Seoul integrate GPS tracking and smart card data for high connectivity (Soh et al., 2010; Yoon et al., 2024). Singapore's rail-bus network, supported by a $1.5 billion infrastructure, achieves an average path length of 1.1 but remains financially inaccessible to smaller cities. Similarly, Seoul's subway data analysis enhances passenger flow but prioritizes rail systems and raises privacy concerns. Budapest's multi-modal network identifies high-traffic routes (84% of traffic on 23% of edges) but suffers from weak integration across transport modes, limiting its adaptability to bus-based networks (Haznagy et al., 2015).

2.2 Bus Tracking and IoT Solutions

IoT-based tracking systems provide real-time location updates but often lack scalability or predictive capabilities. Kassim et al.'s GPS-RFID system achieves reliable localization via GSM/GPRS, though its dependence on costly RFID infrastructure limits widespread use (Kassim et al., 2022). Affordable modules like NEO-6M integrated with microcontrollers reduce costs but experience signal issues in dense urban environments, affecting reliability (Andutan & Ucat, 2022). These solutions struggle to balance affordability and consistent performance in dynamic, infrastructure-limited settings.

2.3 AI-Based ETA Prediction

AI models such as RNN, LSTM, and GRU show promise for the ETA prediction but require significant data and computational resources. RNNs capture temporal trends but face vanishing gradient issues with long sequences (Yuan et al., 2020). LSTMs improve long-term dependency modeling, achieving 3.45-minute MAE, yet demand extensive datasets often unavailable in developing cities (Chughtai et al., 2022). GRUs, while lighter, compromise accuracy for complex traffic patterns (Zargar, 2021). Data augmentation techniques help address scarcity, but the quality of synthetic data can limit model reliability (Iglesias et al., 2023).

2.4 Route Optimization Algorithms

Classical routing algorithms differ in efficiency and applicability. Dijkstra's algorithm efficiently handles static, non-negative graphs but lacks dynamic traffic adaptation (Behún et al., 2022).

A* accelerates pathfinding via heuristics but consumes significant memory (Wang et al., 2024). Bellman-Ford accommodates negative weights but its high computational cost (O(V·E)) renders it unsuitable for real-time urban transit (Arnau et al., 2024). These constraints emphasize the need for lightweight, adaptable algorithms for variable traffic conditions.

2.5 Mapping Platforms

Mapping services facilitate route visualization but present trade-offs. Mapbox offers affordable high-resolution imagery but suffers from inconsistent metadata quality (Jiang et al., 2025). Google Maps API provides live traffic data with notable travel time reductions but incurs high licensing fees (Muñoz-Villamizar et al., 2024). OpenStreetMap's open data is versatile yet demands substantial preprocessing (Li et al., 2023). These limitations hinder their direct adoption in resource-limited environments.

2.6 Security Frameworks

Security remains vital in transit systems managing sensitive data. HTTPS/SSL ensures encrypted communication but introduces certificate management overhead (Nookala, 2024). Multifactor authentication strengthens protection but may reduce usability in low-tech contexts (Wang et al., 2020). Balancing robust security with accessibility is crucial for public transport applications in emerging cities.

In summary, existing public transportation solutions offer advanced features but are often too costly, data-intensive, or infrastructure-dependent for mid-sized cities like Batna. Addressing challenges like unreliable schedules and limited real-time information requires lightweight, affordable, and data-efficient technologies tailored to resource-constrained urban environments.

3 Batna Integrated Transit System

The proposed system adopts a cohesive yet modular architecture that integrates real-time tracking, predictive analytics, and optimal route computation into a unified framework for public bus network management. Leveraging low-cost hardware components and open-source software tools, the system is specifically tailored to operate effectively in resource-constrained urban environments. This section details the system's architecture and operational workflow, encompassing data acquisition through a custom-developed mobile application, preprocessing, model training, and deployment. Real-time GPS data collected from buses serves as the foundation for a machine learning-based ETA prediction module, trained on an augmented dataset to enhance temporal accuracy. Simultaneously, a graph-based shortest path algorithm facilitates efficient route planning. The backend infrastructure and mobile client interface are linked through secure APIs and real-time communication protocols to ensure reliable, low-latency information delivery. Each component has been deliberately chosen and configured to maximize cost-efficiency, scalability, and applicability to similar mid-sized urban contexts.

3.1 System Architecture

The system comprises three modules: real-time tracking, estimated time of arrival (ETA) prediction, and route planning, integrated via a Flask server and Mapbox API (Figure 1). Raspberry Pi 4 and NEO-6M GPS modules transmit location data to a Firebase-backed server. An Android app delivers live tracking, estimated times of arrival (ETA), and route suggestions to commuters.

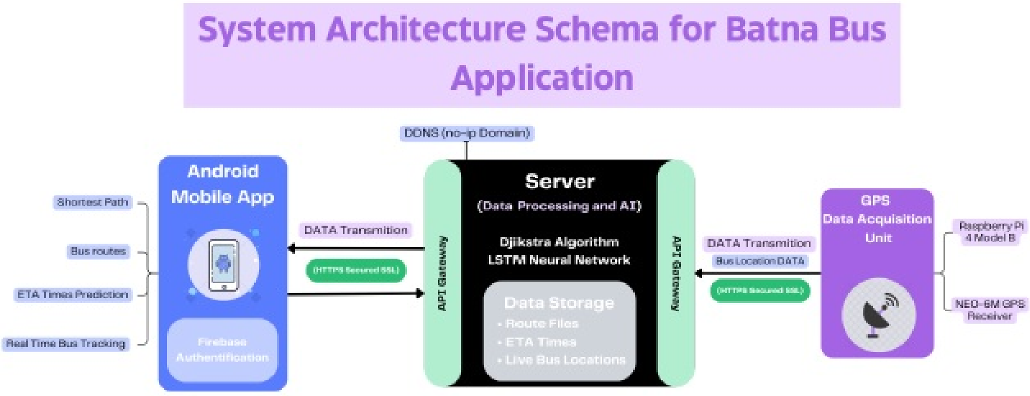

Figure 1: System architecture integrating GPS tracking, LSTM prediction, and Dijkstra routing.

3.2 Data Collection and Preprocessing

Batna's lack of digital public transport data necessitated the development of a dedicated Android application, *Bus Route Collector*, to gather GPS traces from drivers during real bus trips. Each trip recorded both the continuous GPS coordinates of the route and the stop positions with corresponding arrival times. An example of the recorded data format is shown in Table 1.

Table 1: Sample of raw GPS route data recorded by the *Bus Route Collector* app.

Longitude	Latitude
6.16502169	35.53205884
6.16510423	35.53201936
...	...

However, GPS inaccuracies occasionally produced zigzag or erroneous routes, especially in areas with poor signal coverage. To correct these deviations, we used the web-based tool `onthegomap` for route smoothing. This process involved manually redrawing each route based on the recorded GPS traces and accurately repositioning stops. Figure 2 illustrates the improvement achieved through this smoothing process.

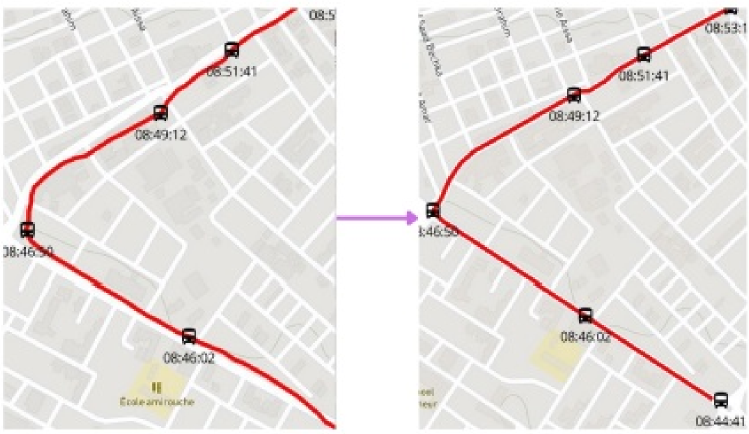

Figure 2: Route data before (left) and after (right) smoothing using `onthegomap.com`.

After refinement, routes and stops were exported as GPX files, converted to CSV, and integrated into the `Route Files` directory for system use.

3.3 Data Augmentation and Preparation

To address the insufficiency of recorded arrival times for LSTM-based ETA prediction, a data augmentation strategy was implemented. Initially, each bus line had only a single trip recording, inadequate for training deep learning models reliant on extended temporal sequences.

We interviewed local bus drivers and consulted the Transport Directorate of Batna to extract operational parameters, including:

- Average trips per bus per day: 5–7.
- Operation hours: 7:00 AM–5:30 PM.
- Average trip duration: 30 minutes.
- Waiting time between trips: 5–15 minutes.

Using these parameters, a Python-based augmentation script simulated five days of bus operations for a fleet of 37 vehicles. Each simulated day randomly assigned 5–7 trips to each bus, introduced random delays to simulate traffic conditions, and included occasional skipped trips to reflect breakdown scenarios. The final augmented dataset included fields such as Day, BusID, TripNo, Direction, Longitude, Latitude, and ArrivalTime.

This augmentation process produced a realistic, diverse, and temporally rich pseudo-dataset (`eta_times`) to feed the LSTM model. The disclaimer remains that while this simulated data reflects actual operational patterns, it will be progressively replaced by live, continuously collected GPS data upon system deployment.

3.4 ETA Prediction Using LSTM Model

To estimate the Expected Time of Arrival (ETA) of buses, a Long Short-Term Memory (LSTM) neural network was developed and trained. LSTM networks are particularly effective for time series forecasting tasks due to their capacity to model long-term temporal dependencies and retain relevant historical information over extended sequences.

The training dataset was constructed through augmentation and comprised simulated bus trajectory data spanning five days. To capture temporal dynamics, the input data was segmented into overlapping sequences of 10 time steps using a sliding window approach. Each time step included eight distinct features representing contextual and spatial attributes of bus movements, thereby enabling the model to infer temporal patterns and behavioral trends.

The architecture of the proposed LSTM model is as follows:

- An input layer configured to process sequences of 10 time steps, each with 8 features;
- A recurrent LSTM layer with 512 memory units to extract temporal dependencies;
- A dropout layer with a dropout rate of 0.2 to mitigate overfitting;
- A dense (fully connected) layer consisting of 64 units with ReLU activation to introduce nonlinearity;
- A final output neuron responsible for predicting the normalized ETA.

Model training was conducted using the Adam optimizer with a learning rate of 0.0005. The Mean Absolute Error (MAE) was employed as the loss function to minimize prediction deviations. Hyperparameter optimization was carried out iteratively through empirical evaluation to achieve a balance between model expressiveness and generalization capability.

The trained model was exported in `.keras` format for seamless integration into the Flask-based backend system, enabling real-time ETA inference. Performance comparisons demonstrated that the LSTM model outperformed both Gated Recurrent Units (GRU) and standard Recurrent Neural Networks (RNN), as further discussed in Section4.

3.5 Route Planning

To enable optimal route calculation within Batna's bus network, the raw GPS route data was transformed into a structured graph. Bus stops were assigned unique identifiers, and stops located within a 50-meter radius were unified to account for overlapping services. Using this processed data, an *adjacency matrix* was constructed, where each node represents a bus stop and each edge represents a direct connection between consecutive stops, weighted by their geodesic distance in meters.

The resulting weighted graph was processed by a Python-based route planner module, where Dijkstra's algorithm is implemented to compute the shortest path between a start and destination stop. Given the same stop may appear in multiple routes, the algorithm considers up to three nearest stops to both the start and destination, exploring multiple candidate paths to improve result reliability.

Additionally, a transfer optimization step identifies the most efficient combination of bus routes covering the computed path, aiming to minimize the number of transfers between different buses. The final computed route includes stop sequences, distances, route names, and transfer points, ensuring clear and actionable navigation guidance for users.

This preprocessing and route planning logic ensured real-time route search capability across Batna's 13 operational bus routes, despite complex overlapping paths and non-symmetrical go/return services.

3.6 Security Measures

To ensure data confidentiality, integrity, and secure access across the system's components, several complementary security mechanisms were implemented.

First, a Dynamic DNS (DDNS) service was configured using No-IP to maintain a stable domain name (`batnabus.ddns.net`) for external access, despite dynamic IP changes on the server network. Port forwarding and firewall rules were applied to allow secure communication through port 8443.

All communications between client devices, including Android apps and Raspberry Pi GPS modules, and the Flask server are encrypted using HTTPS secured with a self-signed SSL/TLS certificate. The certificate was generated using the *win-acme* ACME client and manually distributed to client devices. Certificate pinning was integrated into the Android application, ensuring it only communicates with the authenticated server, effectively mitigating man-in-the-middle attack risks.

User authentication and data access control are managed through Firebase Authentication and Firestore. Authentication relies on email and password credentials, securely hashed using scrypt, with persistent session management provided by the Firebase SDK. Firestore database access is restricted via OWASP-compliant security rules, allowing read and write operations exclusively for authenticated users:

```
allow read, write: if request.auth != null;
```

These layered measures were validated through a security audit involving traffic interception tests and authentication workflow verification, ensuring robust protection for both user data and system communications.

4 Results

The system was evaluated on simulated data, reflecting Batna's 13-bus network.

4.1 ETA Prediction

The LSTM model achieved 99.18% accuracy within ±3 minutes, with a Mean Absolute Error (MAE) of 2.31 minutes and an R^2 score of 0.9913 (Table 2). It outperformed GRU (96.44% accuracy, 4.12 min MAE) and RNN (60.91%, 12.60 min MAE), as well as established benchmarks (Chughtai et al., 2022) (3.10 min MAE) and (As & Mine, 2018) (4.20 min MAE).

Table 2: ETA prediction performance comparison.

Model	Accuracy (%)	MAE (min)	R^2score
LSTM	99.18	2.31	0.9913
GRU	96.44	4.12	0.9901
RNN	60.91	12.60	0.9475

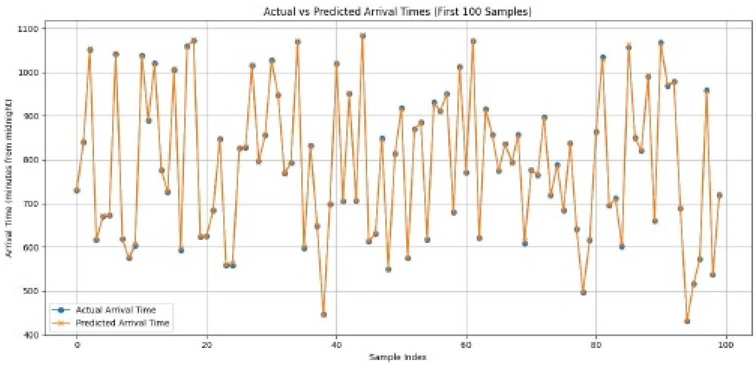

Figure 3: Actual vs. predicted estimated time of arrival (ETA) for LSTM model.

Figure 3 illustrates the performance of the LSTM model by plotting the actual versus predicted arrival times over the first 100 samples. The close alignment between the predicted values (orange line) and actual times (blue line) indicates that the model captures the underlying temporal patterns with high fidelity. Most predictions fall within a narrow error margin, supporting the quantitative metrics reported in Table 2. Minor deviations are observed in regions with more abrupt time shifts, which may correspond to areas with inconsistent bus speeds or simulated traffic conditions. Overall, the figure visually confirms the model's ability to generalize well across sequential input data and accurately forecast ETAs in a realistic, time-dependent transportation scenario.

4.2 Route Planning

Dijkstra's algorithm achieved 100% path correctness and runtimes of 0.67-3.07 ms across 13 routes, 30-145x faster than Bellman-Ford (Table 3). Robustness was 100% under varying conditions.

Table 3: Routing performance comparison.

Algorithm	Correctness (%)	Runtime (ms)	Robustness (%)
Dijkstra	100	0.67-3.07	100
Bellman-Ford	100	95.94-97.29	100

4.3 Security Audit

The system passed OWASP Mobile Top 10 compliance, with end-to-end encryption, certificate pinning, and secure Firestore rules ensuring no unauthorized access.

5 Discussion

The proposed system shows strong potential to enhance the transparency and reliability of public transportation in mid-sized cities like Batna. With 99.18% ETA prediction accuracy and routing times below 3.1 ms, the integration of LSTM models and Dijkstra's algorithm proves effective for real-time transit applications.

Its modular design and reliance on open-source tools make it suitable for resource-constrained environments. The Bus Route Collector app addresses data scarcity by enabling efficient GPS data acquisition during live trips, while the augmentation module enriches the dataset through simulated operational patterns. This combined approach facilitates reliable model training and deployment.

However, current validation is based on simulated data, which may not fully reflect real-world variability. Additionally, the routing algorithm uses static weights, limiting adaptability to dynamic traffic conditions. Limited mobile internet access may also affect adoption.

Future work includes real-world pilot deployment, integration of dynamic traffic data, and extension to additional transport modes. Overall, the system offers a replicable solution for improving urban mobility and advancing digital transport infrastructure in developing cities.

6 Conclusion

This study introduced a real-time bus tracking and route optimization system designed for mid-sized cities. By integrating GPS, LSTM-based ETA prediction, and the Dijkstra algorithm, the system improves the reliability and accessibility of public transport. Its modular, low-cost design makes it adaptable to similar urban contexts. Future work includes pilot deployment and dynamic traffic integration.

References

Andutan, T. A. P., & Ucat, R. C. (2022). Design and Development for a Vehicle Tracking System. *International Journal of Advanced Computer Science and Applications*, *13*(3.

Arnau, R., Martínez, C., & Sánchez, L. (2024). A Bellman-Ford Algorithm for the Path-Length-Weighted Distance in Graphs. *Mathematics*, *12*(16), 2590.

As, M., & Mine, T. (2018). Dynamic bus travel time prediction using an ANN-based model. In *Proceedings of the 12th International Conference on Ubiquitous Information Management and Communication* (pp. 1–8.

Behún, M., Behúnová, A., & Kováč, J. (2022). Recent application of Dijkstra's algorithm in the process of production planning. *Applied Sciences*, *12*(14), 7088.

Chughtai, J., Noulas, A., & Mascolo, C. (2022). An attention-based recurrent learning model for short-term travel time prediction. *PLOS ONE*, *17*(12), e0278064.

Haznagy, A., Fi, I., London, A., & Nemeth, T. (2015). Complex network analysis of public transportation networks: A comprehensive study. In *2015 International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)* (pp. 371–378). IEEE.

Iglesias, G., Talavera, E., & González-Prieto, Á. (2023). Data augmentation techniques in time series domain: a survey and taxonomy. *Neural Computing and Applications*, *35*(14), 10123–10145.

Jiang, H., Ku, M., Zhou, X., et al. (2025). CropLayer: A high-accuracy 2-meter resolution cropland mapping dataset for China in 2020 derived from Mapbox and Google satellite imagery using data-driven approaches. *Earth System Science Data Discussions*, 1–34. https://essd.copernicus.org/preprints/essd-2025-44/

Kassim, M., Ismail, A., & Johari, J. (2022). IoT bus tracking system localization via GPS-RFID. In *2022 IEEE International Conference in Power Engineering Application* (pp. 1–6).

Li, Y., Huang, W., Cong, G., Wang, H., & Wang, Z. (2023). Urban region representation learning with OpenStreetMap building footprints. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (pp. 1363–1373).

Muñoz-Villamizar, A., Faulin, J., Reyes-Rubiano, L., Henriquez-Machado, R., & Solano-Charris, E. (2024). Integration of Google Maps API with mathematical modeling for solving the Real-Time VRP. *Transportation Research Procedia*, *78*, 32–39.

Nookala, G. (2024). The Role of SSL/TLS in Securing API Communications: Strategies for Effective Implementation. *Journal of Computing and Information Technology*, *4*(1).

Soh, H., Lim, S., Zhang, T., Fu, X., Lee, G. K. K., Hung, T. G. G., & Di, P. (2010). Weighted complex network analysis of travel routes on the Singapore public transportation system. *Physica A*, *389*(24), 5852–5863.

Wang, Y., Zhang, Q., & Chen, L. (2024). Multi-Objective Route Planning Model for Ocean-Going Ships Based on Bidirectional A-Star Algorithm. *Applied Sciences*, *14*(17), 8029.

Yoon, J., Kim, S., & Lee, H. (2024). Investigation of Operational Performance of the Seoul Metropolitan Subway Network Using Smart Card Data. *New Physics: Sae Mulli*, *74*, 515–525.

Yuan, Y., Yang, J., & Zhou, X. (2020). Bus Dynamic Travel Time Prediction: Using a Deep Feature Extraction Framework Based on RNN and DNN. *Electronics*, *9*(11), 1876.

Zargar, S. (2021). Introduction to sequence learning models: RNN, LSTM, GRU. Department of Mechanical and Aerospace Engineering, North Carolina State University.

Wang, C., Wang, Y., Chen, Y., Liu, H., & Liu, J. (2020). User authentication on mobile devices: Approaches, threats and trends. *Computer Networks*, *170*, 107118.