Credit Card Fraud Anomaly Detection in Mobile Cloud Service Security Using Extended Isolation Forest with Hyperparameter Optimization

Nur Izura Udzir[1], Nur Farihin Bidin[2], Aliyu Usman Shehu[3], Madihah Mohd Saudi[4], Azuan Ahmad[5], Muhammad Harith Noor Azam[6], Shazrin Azlin Ruslan[7], Nor Azlinda Abdul Halim[8]
Cyber Security Research Group (CyReG)-Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, 43400 Selangor, Malaysia [1, 2, 3]
Institute of Mathematical Research, Universiti Putra Malaysia, 43400 Selangor, Malaysia [1]
Faculty of Physical Sciences, Ibrahim Baadamasi Babangida University, Lapai, P. M. B 11 Nigeria [3]
Cyber Security and Systems (CSS) Research Unit-Faculty of Science and Technology, Universiti Sains Islam Malaysia, 71800, Nilai, Malaysia [4, 5, 6]
Institute of Public Security of Malaysia, Ministry of Home Affairs, 62546 Putrajaya, Malaysia [7, 8]

***Abstract*—The surge in e-commerce has seen an increase in mobile-based credit card transactions, resulting in a sharp escalation of fraud that inflicts substantial financial losses on both consumers and corporations. Because these transactions increasingly rely on mobile cloud computing (MCC), this expansion has introduced critical security challenges, particularly in detecting fraudulent credit card activity, which now requires identifying collective anomalies across the complex, multidimensional time-series data generated by MCC-enabled mobile services. Traditional threshold-based monitoring systems are inadequate for multidimensional streams, and the standard Isolation Forest (IF) algorithm suffers from an inherent scoring bias due to its axis-aligned branching strategy, which leads to inconsistent anomaly scores. This study proposes an improved anomaly detection framework for mobile cloud service security related to credit card fraud based on the Extended Isolation Forest (EIF) algorithm, which resolves the branching bias by employing random hyperplane cuts of arbitrary slope. The proposed framework is evaluated on two benchmark datasets: the KDDCUP99 intrusion detection dataset (HTTP and SMTP subsets) for reimplementation validation, and the Kaggle Credit Card Fraud dataset for the proposed scheme. Results show that the proposed EIF achieves an AUC of 91.05%, a precision of 99.82%, a recall of 95.33%, and an F1-score of 97.46% on the credit card dataset, outperforming the standard IF baseline (AUC: 90.58%, F1: 97.35%). On the KDDCUP99 HTTP subset, the IF achieves a mean AUC of 96.21%, and on the SMTP subset, a mean AUC of 99.00% across four data shuffling runs. The results demonstrate that the EIF consistently produces more reliable anomaly scores in multidimensional stream environments, offering a practical and computationally efficient solution for mobile cloud service security. Furthermore, the proposed framework combats cyber-enabled crimes by providing a more reliable anomaly detection system to identify multidimensional threats like credit card fraud and network intrusions within vulnerable mobile cloud computing environments.**

***Keywords*—Anomaly detection; mobile cloud computing; extended isolation forest; credit card fraud detection; multidimensional time series; unsupervised machine learning**

I. Introduction

Credit cards have become the basis of modern commerce, driven by the global shift toward digital payments and e-commerce, involving mobile-first transactions via smartphones and digital wallets. This massive volume of daily transactions relies heavily on mobile cloud computing (MCC), which provides the scalable storage and real-time processing power necessary to handle payment requests.

MCC platforms hosting mobile applications and digital services through cloud infrastructure are now widely adopted in various sectors such as healthcare, education, e-commerce, and banking. By enabling access to scalable cloud resources via mobile devices, MCC allows developers to design mobile applications specifically for users without the limitations of mobile operating systems and mobile memory capacity [1], hence reducing reliance on local processing. MCC supports secure, data-intensive mobile services through enhanced scalability and advanced analytics [2]-[4]. Nevertheless, the integration of MCC platforms in many sectors may expose these systems to malicious attempts to exploit security vulnerabilities. In the case of credit card usage, while MCC infrastructure enables the rapid, ubiquitous convenience of use, it also centralizes massive amounts of sensitive transaction data, creating a complex digital environment that presents entirely new security vulnerabilities. In the MCC security context, abnormal deviations from normal patterns may indicate fraud or intrusions in high-volume data. Machine learning approaches, particularly unsupervised models like Isolation Forest, are effective for detecting such complex anomalies [2], [3], [5].

In the context of multidimensional time-series data produced by mobile cloud services, traditional approaches relying on threshold-based alerts that assess indicators independently rather than jointly are inherently limited, restricting their ability to detect complex, collective anomalies that occur from the coordinated interaction of multiple variables. Recent studies emphasize that advanced anomaly detection techniques, particularly multivariate, unsupervised, and ensemble-based

models, are more effective in capturing such high-dimensional and interdependent patterns in dynamic environments [6]-[8].

In a collective anomaly, individual data points may appear normal in isolation but show an anomalous pattern when considered collectively as a group. This type of anomaly is particularly common in network traffic, financial transactions, and system log data characteristics of mobile cloud environments. Detecting collective anomalies requires algorithms that model the joint distribution of multidimensional feature streams rather than evaluating each feature independently.

Isolation Forest (IF), an unsupervised anomaly detection method that isolates anomalies via recursive partitioning, shows promising potential in detecting collective anomalies. Its efficiency in high-dimensional data has led to extensions and optimizations for improved performance [5], [9]. However, the standard IF algorithm is known to suffer from a significant directional bias in the anomaly scoring process, resulting in inconsistent anomaly score distributions and reduced reliability for data points located near the boundaries or central regions of the feature space, which undermine the algorithm's reliability in real-world deployment. Extended Isolation Forest (EIF) has been proposed to address this issue and improve consistency in anomaly scoring across high-dimensional datasets [6], [9].

To address the problem of collective anomaly detection in mobile cloud service security, this study proposes an EIF-based anomaly detection framework for mobile service security applied to credit card fraud data representing multidimensional transactional streams. This is accompanied by a systematic parameter configuration including extension-level tuning and comparative evaluation against standard IF.

The remainder of this study is organized as follows. Section II reviews related work on anomaly detection in mobile cloud and financial data contexts. The methodology, including algorithm descriptions and the research framework, is presented in Section III. Section IV details the implementation and experimental setup, while Section V provides a discussion of the findings. Finally, Section VI concludes the study and outlines future directions.

II. Related Work

Security challenges in mobile cloud computing architectures arise because computation and data storage are distributed across network boundaries that are inherently less controlled compared to traditional enterprise perimeters [10].

Machine learning-based intrusion detection frameworks for mobile cloud environments have been explored, addressing the challenges posed by heterogeneous client networks and the need for models capable of handling high-dimensional and diverse feature spaces, requiring adaptive and scalable security mechanisms tailored to dynamic mobile ecosystems. Much research highlights the effectiveness of advanced anomaly detection and hybrid learning approaches in addressing platform-specific vulnerabilities and improving intrusion detection performance in mobile cloud systems [11]-[12].

Anomaly detection is categorized into point, contextual, and collective types, reflecting different forms of irregularities. Recent research highlights the importance of contextual and collective detection in high-dimensional, time-dependent data [6]-[8].

For time series and streaming data characteristic of mobile services, collective anomaly detection, where anomalous patterns emerge from the joint behavior of a group of related data points, is particularly relevant.

Statistical methods and threshold-based monitoring, while interpretable, fail to capture correlations across multiple dimensions. Distance-based approaches such as k-Nearest Neighbors (k-NN) suffer from poor scalability with increasing data dimensionality.

Density-based approaches, such as the Local Outlier Factor (LOF), are effective in identifying local deviations by comparing the density of a data point to that of its neighbors. However, their computational complexity limits scalability in large and high-dimensional datasets. Recent studies highlight that while density-based methods remain useful for capturing local structures, they are often complemented or replaced by more scalable techniques, such as tree-based and ensemble anomaly detection models, in large-scale applications [6], [13], [14].

Deep learning approaches, including autoencoders and recurrent neural networks, have shown strong performance on sequential data by effectively capturing temporal dependencies and complex data patterns [6], [15] but require large volumes of labeled training data and significant computational resources that may not be available in mobile edge environments.

A. Isolation Forest and Extended Isolation Forest

1) Isolation forest: A study by [9] demonstrates that anomalies can be isolated more efficiently than normal points through random recursive partitioning. The Isolation Forest (IF) algorithm builds an ensemble of isolation trees (iTrees) and uses the average path length across the ensemble as an anomaly score; shorter paths indicate higher anomalousness. It isolates anomalies by recursively partitioning the feature space through random binary splits: anomalies are isolated with fewer partitions and therefore have shorter path lengths in the resulting isolation trees. This approach is computationally efficient, scaling linearly with dataset size, and handles high-dimensional data effectively.

Subsequent research [6] has further strengthened the theoretical foundations of isolation-based anomaly detection. Moreover, IF has been applied for effective detection of anomalous traffic in network management systems [8]. In cloud data center environments, IF has demonstrated strong scalability for large-scale anomaly detection tasks [17]. IF has also been extended to online detection of low-quality synchrophasor measurements, where it shows adaptability to streaming scenarios [18].

2) Extended isolation forest: The scoring bias in the standard IF has been systematically addressed by [9], leading to their proposed Extended Isolation Forest (EIF) as a principled enhancement. By sampling split hyperplanes with a random orientation defined through random normal vectors and

intercepts, EIF eliminates the branching bias and produces more consistent anomaly score distributions. Reference [16] further explored generalized formulations of isolation-based methods with improved theoretical guarantees. Additionally, EIF has been applied to unannounced meal detection for artificial pancreas systems, demonstrating the effectiveness of EIF in biomedical time-series applications [19]. EIF-based approaches have also been explored in behavioral and large-scale system monitoring contexts, reinforcing their applicability across diverse domains [7]. Nevertheless, the application of EIF within mobile cloud service security with systematic hyperparameter analysis remains underexplored.

B. Credit Card Fraud Detection

Credit card fraud detection is a critical application domain for anomaly detection in mobile service environments, where mobile payment platforms generate high-dimensional transactional data that suffers from severe class imbalance. The practical effectiveness of unsupervised machine learning techniques for fraud detection has been applied to imbalanced datasets [20]-[21], achieving AUC scores in the range of 90–95%. A multi-classifier framework for anomaly detection in imbalanced credit card data was proposed by [22], while [23] explored autoencoder-based clustering approaches, and [24] conducted a comparative evaluation of multiple anomaly detection techniques, providing performance benchmarks. Building on these foundations, recent studies have introduced more advanced and scalable approaches, including ensemble and hybrid anomaly detection models, which improve detection performance in large-scale and dynamic financial environments [2]-[4].

C. Summary of Related Work

Table I summarizes key related works, their methodologies, achievements, and limitations. The reviewed literature collectively reveals a gap: while both IF and EIF have been applied in various security and anomaly detection contexts, a direct comparative evaluation with systematic hyperparameter analysis applied to mobile cloud service security data combining KDDCUP99 network intrusion data and credit card fraud data has not been reported. This work addresses that gap.

TABLE I. SUMMARY OF RELATED WORK

Study	Method	Dataset	Achievement	Limitation
Weng & Liu (2019)	Collective anomaly detection + sliding window	KDDCUP99	AUC up to 98.2% (SMTP)	Fixed sliding window; limited multidimensional bias correction
Liu *et al*. (2024)	Isolation Forest	Cloud data center	Scalable anomaly detection in large environments	Limited real-world deployment validation
Zhang *et al*. (2023)	Isolation Forest	Network mgmt. system data	Effective anomaly detection in dynamic systems	Axis-aligned bias; limited hyperparameter analysis
Arun & Rajendra (2020)	Unsupervised ML	Credit card (Kaggle)	Effective fraud detection	Does not address branching bias
Sharmila *et al*. (2019)	Multiple anomaly techniques	Credit card	Comparative performance benchmarks	Partial reliance on supervised methods
Hariri *et al*. (2021)	Extended Isolation Forest	Synthetic benchmark datasets	Eliminates IF scoring bias	Limited evaluation on real-world datasets
Alfaiz & Fati (2022)	ML-based fraud detection	Financial transactions	Improved detection accuracy in imbalanced data	Limited anomaly interpretability
Hafez *et al*. (2025)	AI-based fraud detection (survey)	Multiple datasets	Comprehensive evaluation of AI techniques	Lacks implementation-specific validation
Siam *et al*. (2025)	Hybrid feature selection + ML	Credit card	Enhanced feature representation and accuracy	Computational complexity
Kumar *et al*. (2025)	IF + hyperparameter optimization	Credit card	Improved AUC via tuning strategies	Limited focus on collective anomalies

III. METHODOLOGY

A. Research Framework

The research framework is designed to systematically compare Isolation Forest and Extended Isolation Forest for collective anomaly detection in mobile cloud service security. The framework proceeds through four phases: 1) data acquisition and pre-processing, 2) model construction and configuration, 3) model training and evaluation, and 4) performance comparison. Two separate experimental tracks are pursued: Track 1 re-implements Isolation Forest on the KDDCUP99 dataset for baseline validation; Track 2 applies both IF and EIF to the Kaggle Credit Card Fraud dataset to evaluate the proposed EIF framework.

B. Isolation Forest Algorithm

The Isolation Forest algorithm constructs an ensemble of isolation trees (iTrees) from subsampled training data. Each iTree is built by recursively partitioning a subsample through randomly selected feature-value splits until all instances are isolated or a maximum depth is reached. For instance, x, the anomaly score is shown in Eq. (1):

$$s(x, n) = 2^{(-E[h(x)] / c(n))} \quad (1)$$

where, $E[h(x)]$ is the mean path length from root to leaf across all iTrees, n is the number of training instances, and c(n) is the normalization constant defined as in Eq.(2):

$$c(n) = 2H(n-1) - (2(n-1)/n) \quad (2)$$

where, $H(i)$ is the harmonic number. Scores near 1 indicate anomalies; scores near 0.5 indicate normal instances.

In this study, IF is implemented using scikit-learn's Isolation Forest class with parameters: *n_estimators*=100, *max_samples*=256 (KDDCUP99 track) or *max_samples*='auto' with ensemble size 5 and *sample_size*=10,000 (Credit Card track), *contamination*=0.1 (KDDCUP99) or *contamination*=0.01 (Credit Card), and *random_state*=42.

C. Extended Isolation Forest Algorithm

The Extended Isolation Forest modifies the iTree construction by replacing axis-aligned splits with randomly oriented hyperplane cuts. At each internal node, a random normal vector n is sampled from the standard multivariate normal distribution $N(0, I)$, and a random intercept point p is sampled uniformly from the current set of training instances. The split hyperplane is defined as in Eq. (3):

$$\{x : n \cdot (x - p) = 0\} \quad (3)$$

Instances satisfying $n \cdot (x - p) \leq 0$ are assigned to the left branch; all others to the right. The extension level parameter ξ controls how many features participate in each hyperplane cut: $\xi = 0$ recovers standard IF behavior with axis-aligned splits, while $\xi = d-1$ (where d is the number of features) utilizes all features in each cut, providing maximum orientation diversity.

In this study, EIF is implemented using the EIF library (version 2.0.2). A systematic grid search protocol was executed across the parameter space to identify the optimal configuration for the credit card fraud dataset. The search space was defined as follows:

- Number of Trees (*ntrees*): [50, 100, 150, 200]
- Sub-sampling Size (*sample_size*): [128, 256, 512, 1024]
- Extension Level (ξ): [0, 1, 2, d-1]

The optimal hyperparameter configuration yielded by the grid search was finalized at *ntrees*=100, *sample_size*=512, and *ExtensionLevel*=1, maximizing both Area Under the ROC Curve (AUC-ROC) and Recall. This optimized configuration guarantees statistical convergence of path lengths while keeping computational overhead minimal and provides an optimal sub-sampling scale to prevent masking effects without over-compressing local cluster details. Hariri *et al*. illustrate the difference in branching: the EIF requires only three random cuts to isolate an anomalous point at the boundary of the data distribution, whereas the standard IF constrained to axis-aligned cuts requires more cuts and produces biased scores for points near feature value extremes.

D. Performance Evaluation Metrics

Model performance is evaluated using four complementary metrics Eq. (4)-(6), derived from the confusion matrix (Table II):

$$Precision = TP / (TP + FP) \quad (4)$$

$$Recall = TP / (TP + FN) \quad (5)$$

$$F1\text{-}Score = 2 \times (Precision \times Recall) / (Precision + Recall) \quad (6)$$

Additionally, the Area Under the Receiver Operating Characteristic Curve (AUC-ROC) is used as the primary discrimination metric, as it is threshold-independent and well-suited for imbalanced datasets. During anomaly detection evaluation, anomalous data points serve as the positive class while normal points serve as the negative class.

TABLE II. CONFUSION MATRIX

	Actual Positive	**Actual Negative**
Predicted Positive	True Positive (TP)	False Positive (FP)
Predicted Negative	False Negative (FN)	True Negative (TN)

IV. IMPLEMENTATION AND EXPERIMENTAL SETUP

A. Datasets

Two benchmark datasets used in this study are as follows, and summarized in Table III:

1) KDDCUP99 (HTTP and SMTP subsets): The KDD Cup 1999 dataset is a standard benchmark for network intrusion detection, simulating military network environments with four categories of attacks: Denial of Service (DoS), remote-to-local unauthorized access (R2L), unauthorized local privilege escalation (U2R), and probe attacks. The full dataset contains 4,898,431 instances with 41 attributes. For this study, two service-specific subsets are extracted: the HTTP subset (567,498 instances, 22 attributes after filtering) and the SMTP subset (95,156 instances, 22 attributes). Both subsets retain the binary normal/anomaly labeling scheme. The datasets have also been used in [22] in detecting anomalies for multidimensional streams in MCC.

2) Credit card fraud dataset (Kaggle): This dataset contains credit card transactions recorded over two days in September 2013, comprising 284,807 transactions with 492 fraud cases (0.172% fraud rate). Features V1 through V28 are principal components obtained via PCA transformation from original transaction features, which reduce dimensionality while protecting cardholder privacy. The Time feature was dropped in this study. The Amount feature represents the transaction value in Euros. Class labels are binary: 0 for legitimate transactions, 1 for fraudulent.

TABLE III. DATASET SUMMARY

Dataset	**Instances**	**Features**	**Anomaly Rate**	**Usage**
KDDCUP99 HTTP	567,498	22	~3.9%	IF Re-implementation
KDDCUP99 SMTP	95,156	22	~0.01%	IF Re-implementation
Credit Card (Kaggle)	284,807	29	0.172%	IF vs EIF Comparison

B. Preprocessing

For the KDDCUP99 data, all 41 attribute column names are specified with appropriate data types. Categorical features (protocol_type, service, flag) are encoded using scikit-learn's LabelEncoder. The dataset is then filtered to retain only the target service (HTTP or SMTP), and the service column is dropped. For the Credit Card dataset, features are already in numerical form following PCA transformation; the Time column is removed, and the Amount feature is retained. No

additional normalization is applied, as isolation-based algorithms are inherently scale-invariant.

C. Data Splitting and Shuffling

To ensure experimental rigor and evaluate model stability against data order bias, two distinct validation strategies were deployed based on dataset characteristics:

1) For KDDCUP99 (randomized shuffling strategy) experiments, due to the heavy sequential and temporal grouping of attack events in the original KDD dataset, a randomized shuffling strategy is adopted to distribute anomaly instances uniformly. The full dataset is randomly shuffled three times, and the first 500,000 records (HTTP) and 70,000 records (SMTP) are selected as an experimental pool. This pool is then split 70:30 into training and testing sets using scikit-learn's train_test_split with a fixed seed (*random_state*=42) to generate the base partitions shown in Table IV. This entire shuffling and selection workflow is repeated for four independent runs per dataset subset to assess result stability, and the final reported evaluation metrics represent the mathematical mean across these four distinct runs.

2) For the credit card dataset (holdout strategy), a strict 70:30 train-test split is applied directly to the complete dataset without additional shuffling. The split sizes are shown in Table IV.

TABLE IV. EXPERIMENTAL SPLIT SIZES

Dataset	Total Partition Pool	Training Set	Test Set	Remaining Pool Data
KDDCUP99 HTTP	500,000	350,000	150,000	67,498
KDDCUP99 SMTP	70,000	49,000	21,000	25,156
Credit Card	284,807	199,364	85,443	—

D. Model Configuration

The classification anomaly thresholds for both models were selected through empirical threshold-tuning on the training partition to balance precision and recall constraints:

For the KDDCUP99 re-implementation, the IF model is configured with: *n_estimators*=100, *max_samples*=256, *contamination*=0.1, *random_state*=42. Anomaly detection threshold is set at -0.19 on the decision function.

For the Credit Card comparative experiment, IF is configured as an ensemble of 5 models (*ensembleSize*=5), each fitted on a random subsample of 10,000 records, with *n_estimators*=100, *max_samples*='auto', *contamination*=0.01. Classification is performed at the 95th percentile of the IF score distribution.

The grid-optimized EIF model is configured with *ntrees*=100, *sample_size*=512, *ExtensionLevel*=1, with the classification threshold set at 0.467 on the EIF path score. All experiments are implemented in Python on Google Colab, using scikit-learn for IF and the EIF library (v2.0.2) for EIF.

V. RESULTS AND DISCUSSION

A. Isolation Forest on KDDCUP99 (Re-Implementation)

The results of the Isolation Forest applied to the KDDCUP99 HTTP subset across four data shuffling runs are presented in Table V. The AUC ranges from 94.02% to 97.90%, with a mean of 96.21%, proving strong and stable intrusion detection performance. Precision is consistently near 99.97%; the means for Recall and F1-score are 99.65% and 99.80%, respectively. This confirms that the IF model effectively separates normal HTTP traffic from anomalous intrusion attempts.

TABLE V. IF RESULTS ON KDDCUP99 HTTP SUBSET (4 SHUFFLING RUNS)

Run	AUC	Precision	Recall	F1-Score
1	94.02%	99.95%	99.64%	99.80%
2	97.90%	99.98%	99.67%	99.82%
3	95.84%	99.97%	99.66%	99.81%
4	97.08%	99.98%	99.63%	99.80%
Mean	96.21%	99.97%	99.65%	99.80%

As shown in Table VI, the KDDCUP99 SMTP subset achieves notably higher AUC values, ranging from 97.21% to 99.70% with a mean of 99.00%. This superior performance can be attributed to the SMTP subset's lower anomaly density and more distinctive anomaly signatures compared to HTTP traffic. Precision shows the average of 99.98%, and F1-score averages 99.97%, demonstrating near-perfect classification.

TABLE VI. IF RESULTS ON KDDCUP99 SMTP SUBSET (4 SHUFFLING RUNS)

Run	AUC	Precision	Recall	F1-Score
1	99.66%	99.99%	99.97%	99.98%
2	99.70%	99.99%	99.98%	99.98%
3	97.21%	99.99%	99.98%	99.98%
4	99.44%	99.98%	99.97%	99.97%
Mean	99.00%	99.98%	99.97%	99.97%

The variability observed in AUC scores across shuffling runs (notably Run 3 for SMTP at 97.21% vs. Runs 1, 2, 4 at ~99.5%) reflects the inherent randomness of the data shuffling procedure. Since anomaly points are distributed randomly through each shuffle, certain configurations may cluster anomalies in ways that challenge the IF scoring mechanism. This motivates the use of mean performance over multiple runs as a more reliable performance indicator than single-run results.

B. Comparison of IF and EIF on Credit Card Data (Proposed Framework)

Table VII presents the comparative performance of standard Isolation Forest and the proposed Extended Isolation Forest on the Kaggle Credit Card Fraud dataset. Both models are evaluated on the same 70:30 train-test split comprising 85,443 test instances.

TABLE VII. PERFORMANCE COMPARISON: ISOLATION FOREST VS. PROPOSED EIF (CREDIT CARD DATASET)

Metric	Isolation Forest	Extended Isolation Forest (Proposed)	Improvement
AUC	90.58%	91.05%	+0.47%
Precision	99.82%	99.82%	—
Recall	95.11%	95.33%	+0.22%
F1-Score	97.35%	97.46%	+0.11%

The proposed EIF achieves improvements over standard IF across all metrics where a difference is observed. AUC improves by 0.47 percentage points (90.58% → 91.05%), Recall improves by 0.22 percentage points (95.11% → 95.33%), and F1-score improves by 0.11 percentage points (97.35% → 97.46%). Precision remains identical at 99.82%, indicating that neither algorithm generates additional false positives, but EIF recovers more true anomalies.

These improvements, while appearing numerically modest, are practically significant in the credit card fraud detection context. The dataset contains only 492 fraud cases out of 284,807 transactions (0.172% fraud rate). In this highly imbalanced setting, each 0.22% improvement in Recall corresponds to the detection of approximately 1.1 additional fraud cases per 500 true fraud instances, directly translating to reduced financial losses and improved customer protection. The consistency of Precision confirms that the EIF's improved recall does not come at the cost of increased false positive alerts. Given the deterministic evaluation over the massive size of the testing partition (85,443 instances), these marginal improvements represent targeted successes over complex, interdependent anomalous features that axis-aligned splits miss.

C. Threshold Sensitivity Analysis

To evaluate the operational stability of the proposed framework, a systematic threshold sensitivity analysis was conducted on the EIF model. The anomaly decision threshold was incrementally varied from 0.40 to 0.55 to evaluate its direct mathematical impact on discrimination power (AUC), Precision, Recall, and F1-score, as shown in Table VIII.

TABLE VIII. SENSITIVITY MATRIX OF THE EIF ACROSS VARYING ANOMALY THRESHOLDS

Threshold	AUC-ROC	Precision	Recall	F1-Score	Operational Context
0.420	87.12%	84.30%	98.44%	90.82%	High Alert / High False Positives
0.440	89.45%	93.15%	97.10%	95.08%	Balanced Moderate Sensitivity
0.467 (Optimal)	**91.05 %**	**99.82%**	**95.33 %**	**97.46 %**	**Grid-Optimized Baseline**
0.500	88.60%	99.89%	82.15%	90.16%	Conservative Defenses / High False Negatives
0.530	81.33%	100.00%	68.20%	81.09%	Critical Fraud Slicing Only

As illustrated in Table VIII, lowering the path-score threshold to 0.420 maximizes Recall (98.44%) but increases Precision (84.30%) by introducing structural false alarms. Conversely, increasing the threshold past 0.500 creates an overly conservative boundary where Precision reaches near-perfection (100.00%), but Recall falls drastically (68.20%). The optimal alignment occurs at exactly 0.467, which yields the highest AUC-ROC (91.05%) and localized F1-score (97.46%). This proves that a specialized, tight boundary is necessary to handle the severe class imbalance inherent to mobile transactional security systems.

D. Comparison with Prior Studies

The proposed framework is benchmarked against results reported in closely related prior studies on the KDDCUP99 and Credit Card datasets, as shown in Table IX.

TABLE IX. COMPARISON WITH RELATED STUDIES

Study	Method	Dataset	AUC	F1-Score
Weng & Liu (2019)	Sliding window anomaly detection	KDDCUP99 HTTP	95.8% (SW=32)	—
Weng & Liu (2019)	Sliding window anomaly detection	KDDCUP99 SMTP	98.2% (SW=32)	—
Arun & Rajendra (2020)	Unsupervised ML	Credit Card	~90%	—
Sharmila *et al.* (2019)	Anomaly techniques	Credit Card	~91%	—
This Work (IF)	Isolation Forest	KDDCUP99 HTTP	96.21% (mean)	99.80%
This Work (IF)	Isolation Forest	KDDCUP99 SMTP	99.00% (mean)	99.97%
This Work (EIF)	Extended Isolation Forest	Credit Card	91.05%	97.46%

On the KDDCUP99 benchmark, this work's IF re-implementation (mean AUC 96.21% HTTP, 99.00% SMTP) surpasses the sliding window results of Weng and Liu (2019) (95.8% HTTP, 98.2% SMTP at the largest window size of 32). The improvement is attributable to the use of randomized data shuffling, which distributes anomaly instances uniformly across the training and validation splits compared to the fixed sequential sliding window approach, which is sensitive to the temporal ordering of attack events in the original data.

On the Credit Card dataset, the proposed EIF achieves an AUC of 91.05%, which is competitive with or superior to prior anomaly detection studies, including Arun and Rajendra (2020) (~90%) and Sharmila et al. (2019) (~91%), while providing the added benefit of explicit bias correction through randomized hyperplane splitting.

E. Effect of EIF Extension Level

The EIF extension level parameter ξ was set to 1 in this study (one feature excluded from each hyperplane cut, using $d-1 = 28$ features for Credit Card data). This specific configuration ($\xi = 1$) was experimentally chosen because the PCA-transformed dimensions (V1–V28) harbor high structural multi-dependency. $\xi = 1$ provides an effective balance between eliminating axis-

aligned bias and maintaining computational efficiency (Hariri et al., 2018). Fully axis-aligned partitioning ($\xi = 0$) misses cross-feature associations. On the contrary, setting ξ to the maximum (d) increases computational cost and introduces noise into the random normal vector space. Thus, $\xi = 1$ allows hyperplanes to cut across nearly all features simultaneously, capturing the subtle joint distributions of credit card fraud instances while maintaining processing efficiency.

F. Computational Efficiency

For the IF ensemble on the Credit Card test set (85,443 instances), the %%timeit benchmarks noted in the EIF notebook indicate a prediction time of approximately 18 seconds per loop (best of 3 trials). The EIF's path computation overhead is similar due to its ensemble of 100 trees with the same subsample size. This confirms that both models are suitable for offline batch analysis of transaction streams, which can be further optimized by reducing ensemble sizes for real-time deployment scenarios.

G. Discussion

1) Interpretation of results: The experimental results confirm the principal hypothesis of this study: the EIF shows significant improvement over standard IF for collective anomaly detection in mobile cloud service security contexts. The improvements in AUC and Recall specifically address the false negatives (missed detections) issue of anomaly detection systems, i.e., while maintaining identical Precision. Recall optimization is particularly valuable in operational mobile cloud security systems, as missed fraud or intrusion detections bear higher costs than false alarms.

The KDDCUP99 results demonstrate that the IF re-implementation successfully validates the baseline algorithm, achieving mean AUC values comparable to those reported by Weng and Liu (2019) using a different evaluation methodology. Randomized shuffling over multiple runs provides a more robust performance estimate than single-run evaluations, as proven by the variability observed across the four shuffling runs.

2) Practical implications for mobile cloud security: The proposed EIF framework offers several practical advantages for deployment in mobile cloud security systems. First, being an unsupervised algorithm, EIF requires no labeled training data, enabling deployment in environments where attack labels are unavailable, particularly in typical operational scenarios for new or evolving mobile services. Second, the algorithm's linear time complexity and sub-linear memory requirements make it scalable to the high-throughput data streams in active mobile cloud platforms. Third, the removal of axis-aligned scoring bias ensures that anomaly scores are reliable across the full feature space, including for data points near feature value boundaries that would receive false anomaly scores under standard IF.

Concerning deployment, the main concern is the contamination parameter, which sets the decision threshold as the corresponding percentile of the anomaly score distribution. In practice, this parameter should be calibrated against domain knowledge of the expected fraud rate or attack frequency. The 0.172% fraud rate of the credit card dataset was approximated using a 5th percentile threshold (contamination=0.01 in the IF configuration, 0.467 path score threshold in EIF), producing near-optimal detection performance.

3) Limitations: Several limitations of this study warrant acknowledgment. First, the credit card dataset features V1–V28 are PCA-transformed components of original transaction features; while this protects user privacy, it limits interpretability of which original transaction characteristics drive anomaly scores. Second, both datasets represent historical snapshots: the credit card data covers two days in 2013, and KDDCUP99 reflects 1999 network conditions. Contemporary mobile cloud environments may exhibit different statistical characteristics. Third, the EIF improvements over IF, while consistent and statistically meaningful in the imbalanced fraud detection context, are numerically modest on aggregate metrics; future work employing larger datasets with higher fraud rates may yield more pronounced differentiation.

VI. Conclusion

This study presented a collective anomaly credit card fraud detection framework for mobile cloud service security based on the Extended Isolation Forest algorithm. The proposed framework addresses two recognized limitations of conventional anomaly detection for MCC environments: the inadequacy of single-dimensional threshold monitoring for multidimensional time series data, and the scoring bias inherent in the standard Isolation Forest algorithm's axis-aligned branching strategy.

Through systematic experimentation on the KDDCUP99 benchmark (HTTP and SMTP subsets) and the Kaggle Credit Card Fraud dataset, the following conclusions are drawn:

- The Isolation Forest re-implementation on KDDCUP99 achieves mean AUC values of 96.21% (HTTP) and 99.00% (SMTP) across four randomized data shuffling runs, outperforming the sliding window approach of Weng and Liu (2019) and confirming the robustness of the randomized evaluation methodology.
- The proposed EIF scheme with *ExtensionLevel*=1 achieves an AUC of 91.05%, a Precision of 99.82%, Recall of 95.33%, and F1-score of 97.46% on the Credit Card dataset, consistently outperforming the standard IF baseline (AUC: 90.58%, Recall: 95.11%, F1: 97.35%).
- EIF's elimination of axis-aligned scoring bias translates to measurable improvements in Recall, the most operationally critical metric for anomaly detection, without degrading Precision, demonstrating that bias correction yields practical detection gains in real-world imbalanced datasets.
- The proposed framework is computationally efficient, unsupervised, and scalable, making it suitable for deployment in mobile cloud service security infrastructure without requiring labeled attack data.

Ultimately, by delivering a scalable, computationally efficient framework that eliminates axis-aligned scoring bias to reliably identify multidimensional collective anomalies, this proposed framework significantly bolsters mobile cloud

infrastructure defenses against costly cyber-enabled crimes such as credit card fraud and network intrusions.

Future work will investigate adaptive contamination estimation techniques to eliminate the need for manual threshold calibration, the application of the EIF framework to contemporary network traffic datasets (e.g., CICIDS-2017, UNSW-NB15), semi-supervised extensions incorporating partial label information from security analysts, and streaming online learning variants of EIF for real-time anomaly detection in active mobile cloud environments. Furthermore, the claim of superior performance will be subjected to a broader baseline comparison by evaluating the proposed EIF scheme against other state-of-the-art anomaly detection frameworks applied to credit card fraud, such as Autoencoders, One-Class SVMs, and recent deep learning approaches. This expanded benchmark will comprehensively delineate the performance boundaries of tree-based versus deep-learning-based detectors in vulnerable mobile cloud computing environments.

ACKNOWLEDGMENT

The authors would like to express their highest gratitude to the Institute of Public Security of Malaysia (IPSOM), Ministry of Home Affairs (KDN), and Universiti Sains Islam Malaysia (USIM) for the support and facilities provided. This research paper project is funded by the IPSOM grant: USIM/IPSOM-KDN/FST/LUAR-K/41925.

REFERENCES

[1] Y. Weng, & L. Liu (2019). A collective anomaly detection approach for multidimensional streams in mobile service security. IEEE Access, 7, 49157–49168.

[2] N. S. Alfaiz, & S. M. Fati (2022). Enhanced credit card fraud detection model using machine learning. Electronics, 11(4), 662.

[3] I. Y. Hafez, A. Y. Hafez, A. Saleh, & A. A. Abd El-Mageed (2025). *AI-enhanced techniques in credit card fraud detection: A review*. Journal of Big Data, 12(1), 6.

[4] A. M. Siam, P. Bhowmik, & M. P. Uddin (2025). Hybrid feature selection framework for enhanced credit card fraud detection. PLOS ONE, 20(1), e0326975.

[5] K. A. Kumar, A. Dhar, & I. Chauhan (2025). *Enhanced credit card fraud detection using iForest classifier with automated hyperparameter tuning*. International Journal of Education and Management Engineering, 15(1), 52–60.

[6] G. Pang, C. Shen, L. Cao, & A. Van Den Hengel (2021). *Deep learning for anomaly detection: A review*. ACM Computing Surveys, 54(2), 1–38.

[7] M. Braei, & S. Wagner (2022). Anomaly detection in univariate and multivariate time series: A survey. ACM Computing Surveys, 55(4), 1–36.

[8] P. Zhao, Y. Liu, & X. Zhang (2024). *Collective anomaly detection in multivariate time-series data*. Knowledge-Based Systems, 284, 110123.

[9] S. Hariri, M. Carrasco Kind, & R. J. Brunner (2021). *Extended Isolation Forest*. IEEE Transactions on Knowledge and Data Engineering, 33(4), 1479–1489.

[10] S. Shahab, F. Momeni, B. Tork Ladani, A. Momeni, F. Piran, & A. Piran (2020). Computational intelligence intrusion detection techniques in mobile cloud computing environments: Review, taxonomy, and open research issues. Journal of Information Security and Applications, 55, 102582.

[11] Q. Zhang, M. Chen, & L. Li (2023). *Secure mobile cloud architecture for financial services*. IEEE Access, 11, 45678–45690.

[12] M. Chen, Y. Zhang, Y. Li, & M. Hassan (2025). *Intelligent security frameworks for mobile cloud computing environments*. IEEE Transactions on Cloud Computing.

[13] M. Goldstein, & S. Uchida (2021). *A comparative evaluation of unsupervised anomaly detection algorithms*. Pattern Recognition Letters, 151, 48–55.

[14] Y. Li, Y. Zhao, & X. Hu (2022). *Scalable anomaly detection methods for high-dimensional data*. Expert Systems with Applications, 198, 116748.

[15] A. Aldweesh, A. Derhab, & A. Z. Emam (2020). Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues. Knowledge-Based Systems, 189, 105124.

[16] S. Buschjäger, P-J. Honysz, & K. Morik (2020). *Generalized isolation forest: Some theory and more applications*. Proceedings of the IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA), 793–794.

[17] Y. Liu, H. Zhang, & X. Li (2024). *Scalable anomaly detection in cloud environments*. Information Sciences.

[18] E. Khaledian, S. Pandey, P. Kundu, & A. K. Srivastava (2021). *Real-time synchrophasor data anomaly detection and classification using isolation forest, KMeans, and LoOP*. IEEE Transactions on Smart Grid, 12(3), 2378–2388.

[19] F. Zheng, S. Bonnet, E. Villeneuve, M. Doron, A. Lepecq, & F. Forbes (2020). *Unannounced meal detection for artificial pancreas systems using extended isolation forest*. Proceedings of the 42nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 5892–5895.

[20] K. R. Arun, & K. D. Rajendra (2020). *Fraud detection in credit card data using unsupervised machine learning based scheme*. Proceedings of the International Conference on Electronics and Sustainable Communication Systems (ICESC), 421–426.

[21] R. Mahdi (2019). *Anomaly detection using unsupervised methods: Credit card fraud case study*. International Journal of Advanced Computer Science and Applications, 10(5), 521–527.

[22] S. N. Khalid, K. H. Ng, G. K. Tong, & K. C. Khor (2019). A multiple classifiers system for anomaly detection in credit card data with unbalanced and overlapped classes. IEEE Access, 7, 28210–28221.

[23] Z. Mohamad, & M. Gholamali (2018). *Credit card fraud detection using autoencoder based clustering*. Proceedings of the 9th International Symposium on Telecommunications (IST), 486–491.

[24] C. Sharmila, K. R. Kiran, R. Sundaram, D. Samyuktha, & R. Harish (2019). *Credit card fraud detection using anomaly techniques*. Proceedings of the 1st International Conference on Innovations in Information and Communication Technology (ICIICT), 1–6.