Journal of Trends in Computer Science and Smart Technology (ISSN: 2582-4104)
www.irojournals.com/tcsst/

CAAMF: Conflict-Aware Adaptive Multimodal Fusion for QR Code Phishing Attack Detection

Nidhi Nigam[1*], Rajat Bhandari[2]

[1]Research Scholar, Department of Computer Science and Engineering, SAGE University, Indore, India.
[2]Professor, Institute of Advance Computing, SAGE University, Indore, India.

E-mail: [1*]nigam.nidhi07@gmail.com, [2]hoi.iaccore@sageuniversity.in

Orcid ID: [1*]0009-0008-9999-0476, [2]0000-0002-4163-9391

Abstract

QR codes are widely utilized in digital transactions; however, substantial security risks are increased by QR-based phishing (quishing) attacks. Existing attack detection techniques depend on a single modality-either analysis of URLs or visual inspection of QR codes causing high false-positive rates. Furthermore, attackers exploit mechanisms that bypass either of these techniques. This paper proposes CAAMF, a Conflict-Aware Adaptive Multimodal Fusion framework which combines different layers for URL risk assessment, visual integrity analysis, and conditional anomaly resolution into a unified pipeline. The URL scoring layer utilizes LightGBM with threat intelligence, the visual layer uses lightweight Convolutional Neural Networks (CNN) to detect tampering, and Isolation Forest acts as a conditional conflict resolver among disagreements between detection modalities. The results demonstrate an accuracy of 95.9 %, an AUC of 0.982, and a false-positive rate of 2.4 %, outperforming the evaluated baseline models. The conflict-aware adaptive fusion framework is significantly optimized for deployment in real-time (less than 25 ms inference) applications in UPI payment and enterprise security.

Keywords: QR Phishing Detection (Quishing), Multimodal Security, Adaptive Fusion, Conflict-Aware Detection, UPI Payment, Enterprise Security.

1. Introduction

The widespread use of Quick Response (QR) codes for digital transactions, authentication, and information distribution has introduced a severe cybersecurity risk: QR-based phishing (quishing) [1]. The attackers incorporate harmful links into the QR code, making use of the trust users place on scanned data, since there is no visible difference between harmful and valid QR codes. Based on current observations, it has been noted that there has been a 400 percent increase in the number of quishing incidents from 2023 through early 2026, ranging from hacked parking meters to harmful business payments [2]. Unlike phishing, quishing avoids URL checking and thus is not detected by email/web filters [3].

*** Corresponding Author**

DOI: https://doi.org/10.36548/jtcsst.2026.3.010
Received: 09.05.2026, received in revised form: 13.06.2026, accepted: 04.07.2026, published: 15.07.2026

However, at present, the detection techniques are varied. The URL-based approach uses lexical analysis, threat intelligence, and machine learning classifiers but fails to deal with obfuscated redirects and newly registered domains [4]. Several other works have looked into the visual analysis of QR codes, which uses image processing or deep learning models to detect distortion in alignment patterns, overlays, and module formations [5]. Even though this methodology can be helpful for identifying physical tampering, it usually fails to do so efficiently in practical situations for various reasons [6].

There have been numerous studies on hybrid approaches that combine more than one static fusion techniques, like the voting approach or detection score. The limitation of these methods is that they do not contain any conflict resolution within modalities, making them highly susceptible to false positives and missed adversarial attacks [7].

The problem with static fusion can be solved through weighted averaging using pre-assigned weights for each modality. These weights are static and do not take uncertainty into consideration. Confidence-based gating systems deal with uncertainty, but poor confidence assignment techniques generate inefficiency. The mixture-of-expert system makes use of dynamic weight assignment and needs large amounts of training data.

To overcome these problems, the proposed approach, a conflict-aware multimodal system, combines the following three complementary layers of analysis: URL risk analysis along with visual verification, and conditional anomaly detection of QR code phishing. The proposed method considers ambiguous situations while remaining efficient in real time. Contrary to traditional approaches, the new method introduces adaptive uncertainty and conflict-aware anomaly modulation.

Key contributions from this study in the context of research include a novel adaptive multimodal fusion architecture that combines URL semantics, visual integrity, and conditional anomaly scoring based on conflicts between these modalities. The conflict-aware nature of this system means that it activates in instances where there is a discrepancy between URL-based and visual-based predictions, an approach not previously described in studies. Rather than focusing on the aggregation of confidence scores, the conflict awareness approach seeks to identify semantic conflicts between the heterogeneous modalities. The end-to-end efficiency of this architecture, in that inference times are under 25 milliseconds, allows for integration with mobile payment applications.

2. Literature Review

The use of QR codes for information exchange, digital payments, and authentication systems has led to exposure to various security threats. Phishing attacks on QR codes have emerged as one of the important attack vectors. Various research papers have considered the security issues of QR codes and phishing in areas like URL analysis, visual integrity checks, anomaly detection, and hybrid approaches. This chapter discusses various literature reviews in three major categories: URL-based phishing detection, visual analysis of QR code integrity, and hybrid/anomaly-based approaches, along with an analysis of gaps through a comparative study of the proposed system.

2.1 URL-Based Phishing Detection

URL-based approaches are the most popularly used for detecting phishing attacks. URL-based techniques involve the use of lexical characteristics (like the length of the URL, and the presence and order of special characters), metadata associated with the domain name (age of the domain name, TLDs, registrar), and classification using machine learning algorithms for the detection of potentially malicious URLs [8], [9]. These approaches have quick interpretability, thus making them better suited for real-time use cases.

The machine learning algorithms, such as Gradient Boosting, Random Forest, and Support Vector Machine, have worked well in detecting malicious links through pattern recognition from large datasets. Jin et al. [11] have shown the effectiveness of LightGBM for real-time intrusion detection through heterogeneous features. The use of lexical, HTML, and behavior-based features as part of feature engineering was emphasized by Kustiawan and Ghauth [12], who showed greater accuracy. Sahoo et al. [10] observed the time involved in detecting malicious URLs, obfuscated redirections, or new URLs following valid patterns. Nigam et al. [3] noted that attackers can use layered redirection or URL-shortening services to get around filters like blacklisting or heuristics based on a URL-only approach, which are vulnerable to quishing.

2.2 Visual Integrity Analysis

Visual inspection includes the geometric properties of the QR code, including distortion of alignment patterns, noise, tampering, and overlay. QR code structures have a defined structure, where data modules and patterns are encoded, including finder, alignment, and timing. Manipulation of these patterns can be detected using image analysis methods. Traditional methods included the use of edge detection, pattern analysis, and statistical analysis of pixel distribution. These techniques performed well in controlled conditions but failed to perform under real-world image capturing scenarios. Sarkhi and Mishra [15] highlighted that Convolutional Neural Networks showed better results in learning structural features from QR images. Alace and Çelik [13] employed a lightweight CNN architecture and performed well under balanced datasets to learn discriminative visual features. Minocha et al. [16] introduced an automatic QR Code image identifier which assists users in the classification of malicious or benign QR codes. Ford and Berry [17] highlighted the application of CNN for QR code automation, embedded in email images, and faced overfitting issues and increased complexity at early stages with high accuracy. This shows the problem of using a visual-only model in an unconstrained environment. Kamnounsing et al. [14] carried out a study involving adversarial halftone QR codes, which retained reliability. The efficacy of visual-only detection does not guarantee accuracy and leads to a higher rate of false negatives when used in practice. Differences in the quality of images and lighting, among other factors, reduce the efficiency of detection algorithms. Visual-only detection of QR image phishing attacks is insufficient on its own.

2.3 Hybrid and Anomaly-Based Detection Approaches

Multimodality systems incorporate different detection modalities to increase resilience. For instance, Rivas et al. [18] presented the AP3X system, which is based on AI and incorporates threat intelligence and lightweight classifiers. These are very efficient for fighting existing threats but cannot offer any visualization. In their study, Wairagade and Ranjan [19]

showed that anomaly detection techniques, together with hybrid models, could be used to detect zero-day attacks, as they allow modeling deviances from normal user behavior. Nevertheless, such approaches consume many computational resources and lack explanation of differences between different modalities, including cases where the URL score and visual score contradict each other [7]. Ndia and Njuguna [20] performed a systematic literature review on QR security and found that multimodality approaches were used in only 8% of the papers.Hadi et al. [21] introduced the integration of classifiers to identify phishing attacks and proposed an adaptive learning model in complex environments. To detect variations in normal system behavior, Isolation Forest models and One-Class Support Vector Machines are commonly used. Liu et al. [22] worked on Isolation Forest for anomaly detection, but high-dimensional datasets and scalability were challenging. To better understand the limitations in existing methods, a comparative study in Table 1 is presented below.

Table 1. Comparative analysis of existing QR phishing detection approaches with limitations

Study	Features/Model	Limitations	Accuracy (in percentage)
[12]	Lexical URL Features	Visual manipulated QR codes cannot be detected	99.45%
[13]	Lightweight Convolution Neural Network Model	Malicious embedded URLs detection issues	95.89
[15]	Deep Learning URL classifiers	Achieves high accuracy but specifically malware detection is challenged	99
[16]	Vision based Features	Performance degradation with blur, noise and lighting variations.	98.13
[18]	Hybrid Boosting Model	Implementation was based on simulation so performance of zero day scenarios or unseen attacks is unproven.	>96
[20]	Hybrid Ensemble Model	Performance overhead issues	97.79

The critical analysis, based on observations and literature, emphasized research gaps where performances in regulated settings are satisfactory but fail in real-world scenarios using single modalities by incorporating URL features or visual interpretation. Hence, hybrid approaches attempted to overcome these limitations, whereas most of them rely on static fusion or non-conflict-based scenarios. This research aims to work on conflict-aware fusion in an adaptive environment to overcome the limitations of these research gaps.

Despite the improved efficiency achieved in hybrid and anomaly-based detection tasks for sophisticated attacks, some limitations persist. Hybrid detection follows static fusion mechanisms, which generate average classifier output or an average scoping system. There are no provisions available to handle conflicts. On the other hand, extensive computational overhead restricts productivity and handling in real-world applications like mobile payments. To address such limitations, the demand for dynamic conflict-aware multimodal detection systems capable of resolving conflicts has arisen.

Cybersecurity researchers significantly emphasized multimodal detections, leveraging additional information such as network traffic, visual appearance, threat intelligence, and behavioral patterns as key factors to improve accuracy. This intersection of heterogeneous data sources is capable of working with complex patterns and real-time applications like fraud identification, malware analysis, and intrusion detection. However, key challenges include increased computational overhead when all modules work on the same data and produce conflicting outputs. Therefore, efficient architectures are needed to engage with real-time applications and overcome identified gaps with robust and scalable outcomes. First, regarding the absence of multi-modality, many studies combine URL and visual features, but none of

them use a tri-layer system that comprises URL, visual, and anomaly features [4], [20]. Second, concerning conflict-resolution structures, existing systems lack explicit conflict addressing among detection results. A conflict-aware multimodal fusion framework combines three analytical layers: URL risk assessment, visual integrity analysis, and conditional anomaly detection, which are fit for real-time applications with upgraded performance. Third is the lack of vulnerability assessment, where single modalities in URL and visual models are vulnerable to multiple parameters. The proposed CAAMF architecture performed well in addressing these gaps with real-world robustness and performance through mathematical implementations [16] - [18].

3. Threat Model and Problem Definition

A threat model scenario is considered where an attacker's objective is to exploit QR code redirects, leading users to fraudulent or harmful web resources. Adversaries have the capability to plan attacks accordingly, encoding malicious URLs and generating new or manipulating existing QR codes to place them in public places. The adversary also has the right to apply techniques to QR codes to obscure their destination and evade detection. They can also tamper with the visual structure of QR codes to make detection more complex.

The proposed CAAMF framework considers the successful scanning and decoding of QR codes and extracts URLs for analysis in real-time environments. After systematic calculations from three analytical layers, the system will be able to classify the QR code as benign or malicious. Detection can be observed for malicious URLs and visual tampering. The framework resolves conflict-aware adaptive fusion for multiple modalities by maintaining false-positive rates and accuracy under real-time scenarios. The following notations are used to represent mathematical expressions:

- I: QR image
- U: Decoded URL
- S_{URL}: URL risk score
- S_{Visual}: Visual risk score
- Δ: Conflict score
- τ: Disagreement threshold
- λ: Fusion weight

Consider the QR sample represented as

$$x = (I, U)$$

The objective is to calculate

$$P(y \mid x), \quad y \in \{0, 1\}$$

Where y=1 represents malicious and y=0 represents benign.

4. Methodology

Unlike traditional hybrid systems that work on static strategies, the proposed conflict-aware multimodal framework follows probabilistic multimodal fusion integrated with conditional conflict-aware anomaly detection. The primary objective is to identify manipulation in URL redirection and visual integrity by maintaining computational efficiency in a real-time environment [7, 20]. Figure 1 below demonstrates the overall workflow of the proposed framework, which represents a modular pipeline in which QR input is handled in parallel by the URL and visual analysis modules, and a fusion layer is used to coordinate the findings of both, estimate modality reliability, and conditionally invoke a conditional anomaly resolver adaptively.

4.1 System Overview

Conflict-Aware Adaptive Multimodal Fusion framework consist layered architectures as

1. URL Risk Assessment Layer
2. Visual QR Integrity Analysis Layer
3. Conflict-Aware Adaptive Multimodal Fusion Layer

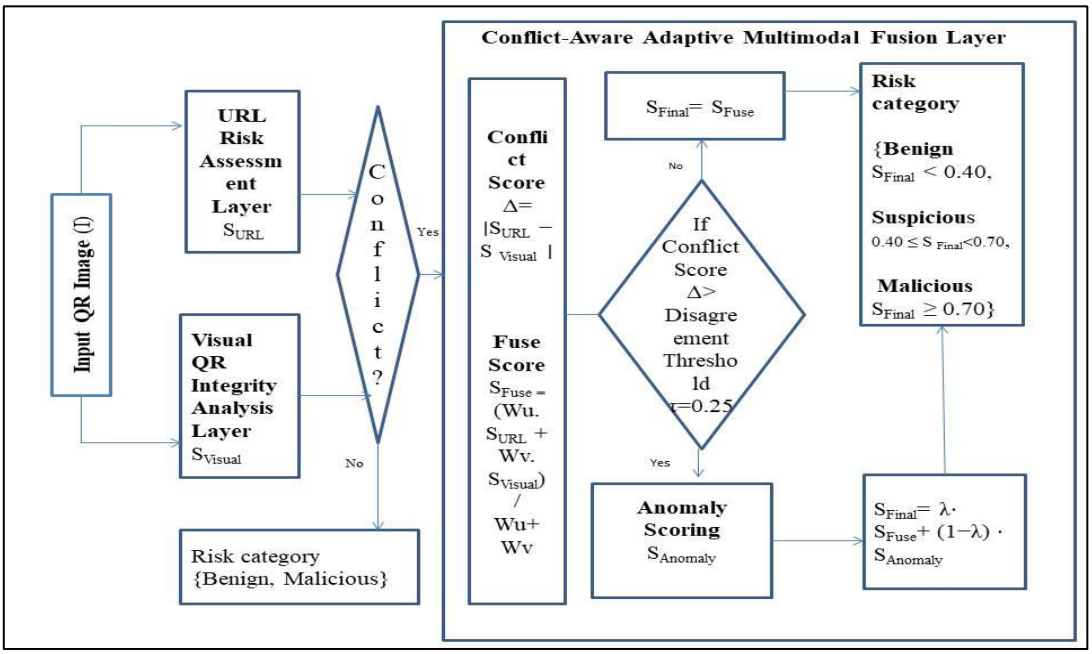

Figure 1. Process flow of proposed CAAMF framework

A QR code scanner first scans the input QR image and extracts the embedded URL. Due to different generation patterns and image transformations, a single QR image maps to a single embedded URL, but multiple QR codes may encode the same URL. A unique mapping process was performed, consisting of approximately 190,000 unique URLs; there were no duplicate URLs within the specific dataset. This URL is further analyzed using structural, lexical, and reputation-based features. Concurrently, the input QR code is processed through a Lightweight Convolutional Neural Network with hyperparameters of 100 trees, a 0.05 learning rate, and a maximum depth of 8 for pattern observations [23, 24].

Conflict is identified by the individual processing of URL and visual features. If both declare the input QR code benign or malicious, the risk category will be displayed without any trigger. The conflict-aware probabilistic fusion mechanism integrates outcomes from both URL and visual modules. A conditional anomaly detection module activates if disagreement between modalities exceeds a predefined threshold. Resources are dynamically allocated for computation while maintaining high accuracy.

To avoid any overlap in train and test sets, URL canonicalization and deduplication were performed before splitting, while duplicate QR codes were detected through perceptual image hashing. Stratified sampling on the domains was done to allocate URLs from the same root domain into one split. After splitting, augmented variations of the QRs were created for use in the train set.

4.1.1 URL Risk Assessment Layer

URL Risk Scoring (Layer 1) computes the likelihood of a decoded URL being malicious by employing an ensemble approach. To confirm the malicious nature of URLs encoded in QR codes, the first layer extracts and combines features from different categories such as lexical features (entropy, percentage of special characters, length), domain metadata (age, registrar's reputation), structural features (subdomain depth, presence of an IP address), and heuristic features (suspicious TLD indicators, redirection chain length). It also considers similarity to threat intelligence feeds like OpenPhish, PhishTank, and URLhaus [23, 24, 26]. These feeds were used as supplementary features rather than a primary detection approach, specifically to avoid over-relying on known repositories.

Table 2 summarizes the features that have been engineered. The extracted features are processed using the LightGBM [11] classifier because it is efficient at dealing with large datasets and heterogeneous feature sets.

Table 2. Engineered URL features for risk assessment

Feature Category	Specific Features	Description
Lexical Features	URL length, Character entropy, Special character density (%, @, //), Digit-to-letter ratio	Statistical properties of the URL string
Domain Metadata	Domain age (days), Registrar reputation score, TLD risk index	WHOIS-based temporal and reputation features
Structural Properties	Subdomain depth, Path length, IP address presence, URL shortening flag	Hierarchical and network-level characteristics
Heuristic Rules	Suspicious keyword count (login, verify, update), Redirection chain length, HTTPS certificate validity	Rule-based threat indicators
Threat Intelligence	OpenPhish match (binary), PhishTank match (binary), Historical phishing count	Real-time and historical threat feed integration

An additional experiment to ensure performance was conducted within a framework both with and without external repositories. The observation shows the system's strong generalization capability.

The assessed URL score is calculated using the formula below:

$$S_{URL} = \alpha \cdot S_{ML} + \beta \cdot S_{Heuristic} + \gamma \cdot S_{Threat} \quad (1)$$

The S_{URL} risk score is calculated as a weighted average of three sub-scores, a machine learning classifier output (S_{ML}), a normalized heuristic score ($S_{Heuristic}$), and a threat intelligence

match score (S_{Threat}). The weights α=0.55, β=0.25 and γ=0.20 are optimized through cross-validation. The generated score is then redirected to the CAAMF layer, where it is combined with the visual analysis score and conflict-aware mechanism.

4.1.2 Visual QR Integrity Analysis Layer

This layer calculates the pattern observations and consistency of a QR code to identify tampering or manipulation. The following steps are performed for such identification:

- QR images are normalized and resized through grey scale conversion into 224*224
- Feature extraction

Feature extraction is to be done by either of two approaches: handcrafted structural features or deep visual features. The layer is based on a hybrid approach to feature extraction, still using handcrafted measures, which are edge asymmetry (measured using the Sobel gradient variance), finder pattern distortion (measured as the Euclidean deviation of ideal alignment), and module regularity, as well as learned representations by a custom lightweight convolutional neural network (CNN). The CNN has three depthwise separable convolutional blocks with 3 × 3 kernels, followed by batch normalization and using the ReLU6 activation function, 50 epochs, and a learning rate of 0.001. It has about 0.4 million parameters. It is optimized for fast inference and runs analysis in less than 12 milliseconds with an estimated computational complexity of 30 MFLOPs (Million Floating Point Operations) with a model memory footprint of 1.6 MB on the test hardware, which fits within mobile deployment specifications mentioned in [15], [25]. To mitigate overfitting, a droprate of 0.3 is employed before final classification. Max-pooling layers reduce spatial dimensionality and preserve discriminative QR structural features. Table 3 below depicts the complete CNN architecture.

Table 3. CNN architecture

Layer	Configuration	Output Size
Input	224×224×1 QR Image	224×224×1
Convolutional Block 1	Depth wise Separable Convolutional (3×3, 32 filters) + Batch Normalization + ReLU	112×112×32
Max Pooling	2×2	56×56×32
Convolutional Block 2	Depth wise Separable Convolutional (3×3, 64 filters) + Batch Normalization + ReLU	56×56×64
Max Pooling	2×2	28×28×64
Convolutional Block 3	Depth wise Separable Convolutional (3×3, 128 filters) + Batch Normalization + ReLU	28×28×128
Global Average Pooling	GAP	128
Dense Layer	128→64 + ReLU	64
Dropout	Rate = 0.3	64
Output Layer	Sigmoid	1

Additionally, EfficientNet-B0 is evaluated as a baseline model under a similar experimental setup to assess the trade-off between detection performance and inference latency.

The output probabilities of CNN tampering are denoted by the visual risk score, S_{Visual}, and P indicates the probability to be predicted using the CNN model. Visual tampering is expressed by the below expressions and maintains low computational overheads.

$$S_{Visual} = P\ (tampered|\ CNN) \quad (2)$$

Similarly, the generated score is forwarded to the CAAMF layer for further processing.

4.1.3 Conflict-Aware Adaptive Multimodal Fusion Layer

Conflict-Aware Adaptive Multimodal Fusion (CAAMF) is proposed to integrate heterogeneous features. Here, the model performs probabilistic fusion for combining modalities through the expressions below: $S_{URL}= (P(y=1 \mid U))$

$$S_{Visual}= (P(y=1 \mid I))$$

To calculate the difference between modalities, Conflict Score Δ is defined which is estimated as

$$\Delta=|S_{URL} - S_{Visual}| \quad (3)$$

Here, S_{URL} and $S_{Visual,}$ individually calibrated with probabilistic outputs from separate features calculus. In contrast to the monotonic agreement between confidences scores assumed in traditional confidence-based ensemble learning, like voting, stacking or weighted averaging, CAAMF highlights underlying strengths rather than noise.

To avoid unreliable modalities, adaptive weights are computed as inversely proportional to uncertainty, which is given as

$$Wm = \frac{1}{(Um\ +\ \epsilon)}$$

Whereas

$$Um = Sm\ (1 - Sm)$$

$Sm \in [0, 1]$, min-max scaling, calibrated maliciousness probability and ϵ is small stabilization constant.

This represents low uncertainty for highly confident predictions and higher uncertainty near the decision boundary. High disagreement between both URL and visual modality doesn't imply a semantic conflict until accompanied by uncertainty levels. Thus, uncertainty conflict Um is introduced to balance the conflicts robustly. Further, adaptive fusion and anomaly resolution layers are activated under semantically ambiguous conditions, which justifies conflict-aware reasoning rather than simple threshold-triggered classification. A fuse score is then calculated.

$$S_{Fuse}=(Wu.\ S_{URL} + Wv.\ S_{Visual})/(Wu+ Wv) \quad (4)$$

Whereas, Wu and Wv are adaptive weights of URL analysis and visual integrity analysis.

QR Sample varies dynamically according to their modality and reliability. Thus, to balance adaptive confidence estimation, it is included in the fusion framework where confidence values will be derived using entropy-based uncertainty calibration as below.

$$Cm=1-Um$$

But when the disagreement levels beyond an empirically acceptable threshold $\tau = 0.25$, the system causes the anomaly detection layer to eliminate the ambiguity. Optimization of the disagreement threshold τ occurred on the validation set by grid search, maximizing the F1-score in trade-off with the rates of false positives and false negatives. The analysis of $\tau \in [0.1, 0.5]$ was performed in steps of 0.05. The trade-off of $\tau = 0.25$ was the best, decreasing the number of unnecessary invocations of the anomaly-detecting procedure by 68% while providing the same value compared to the case when the anomaly-detecting procedure is always invoked ($\tau = 0$). The table below defines the complete threshold sensitivity analysis for disagreement threshold τ.

Table 4. Sensitivity analysis for disagreement threshold τ

Disagreement Threshold τ	Accuracy	F1- Score	False Positive Rate	Anomaly Activation Rate
0.1	0.941	0.943	0.039	42%
0.15	0.948	0.949	0.033	31%
0.2	0.953	0.954	0.028	24%
0.25	0.959	0.956	0.024	18%
0.3	0.954	0.952	0.027	12%
0.4	0.947	0.945	0.034	7%

The selection of fusion weight λ=0.4 is theoretically motivated for the best coordination between deterministic fuse score and uncertainty-aware anomaly detection, as shown in the table below. This conditional activation guarantees the availability of computational resources for truly ambiguous cases, e.g., when a URL is considered benign, but the QR image contains indications of manipulation—a situation not explicitly considered in earlier hybrid systems [13], [21].

Table 5. Sensitivity analysis for fusion weight λ

Fusion Weight λ	Accuracy	F1-Score	ROC-AUC
0.2	0.948	0.949	0.976
0.3	0.954	0.953	0.979
0.4	0.959	0.956	0.982
0.5	0.953	0.952	0.978
0.6	0.947	0.945	0.974

Conditional Anomaly Detection is enabled only when $\Delta > \tau$. This layer uses an Isolation Forest model [22] trained on a combined feature vector that comprises the URL and visual scores, QR structural metrics, redirect-chain entropy, and domain lifecycle anomalies. The model provides an aberration score, $S_{Anomaly}$, which is the min-max scaling value Sm of the sample being an outlier in the learnt distribution of benign patterns. The expression below benign patterns.

$$S'_m = \frac{S_m - \min(S_m)}{\max(S_m) - \min(S_m)}, m \in \{\text{URL, Visual }\}$$

The resulting score of the final decision, S_{Final}, is then determined as a weighted sum

$$S_{\text{Final}} = \begin{cases} \lambda S_{\text{Fuse}} + (1-\lambda) S_{\text{Anomaly}}, & \Delta > \tau, \\ S_{\text{Fuse}}, & \Delta \leq \tau. \end{cases} \quad (5)$$

The values S_{URL}, S_{Visual} and $S_{Anomaly}$ are normalized before fusion to the range [0, 1]. The LightGBM classifier yields regulated probabilities; the CNN outputs sigmoid-based probabilities, whereas Isolation Forest uses min-max scaling.

When no conflict is identified ($\Delta \leq \tau$), the end score is simplified to $S_{Final}=S_{Fuse}$. Isolation Forest [22] was selected over other anomaly detectors because it is computationally efficient andcan be used in cybersecurity. The final score is mapped to one of three actionable classes according to the rule set given below: $S_{Final} \geq 0.70$ is Malicious; $0.40 \leq S_{Final} < 0.70$ is coded as Suspicious to be reviewed further; and samples with a score of less than 0.40 are considered benign.

Algorithm 1: Conflict-aware adaptive multimodal fusion framework

Input: QR image I
Output: Risk category {Benign, Suspicious, Malicious}
U← decode (I)
S_{URL}← URLRiskScoring (U) ▷ **Eq. (1)**
S_{Visual}← VisualScoring (*I*) ▷ **Eq. (2)**
if $S_{URL} < 0.40$ and $S_{Visual} < 0.40$ then
    return Benign
else if $S_{URL} \geq 0.70$ and $S_{Visual} \geq 0.70$ then
    return Malicious
end if
$\Delta \leftarrow |S_{URL} - S_{Visual}|$ ▷ **Eq. (3)**
$Wm \leftarrow 1/ (Um + \epsilon)$
whereas
$Um \leftarrow Sm (1-Sm)$
$S_{Fuse} \leftarrow (Wu.S_{URL} + Wv. S_{Visual})/ Wu + Wv$ ▷ **Eq. (4)**
Set $\tau = 0.25$
If $\Delta > \tau$ then
$S_{Anomaly}$← IsolationForest (U, I)
Set $\lambda=0.4$
$S_{Final} \leftarrow \lambda \cdot S_{Fuse} + (1-\lambda) \cdot S_{Anomaly}$ ▷ **Eq. (5)**
else
    $S_{Final} = S_{Fuse}$
end if
    if $S_{Final} < 0.40$ then
    return Benign
else if $S_{Final} < 0.70$ then
    return Suspicious
else
    return Malicious

The Anomaly Resolver mechanism was activated during testing when approximately 18% of samples exceeded the disagreement threshold ($\Delta > \tau$). Once the exceeded threshold is detected, the system observes modality reliability estimation, adaptive weighting, conditional anomaly assessment, and context-aware decision fusion as additional reasoning. Hence, the disagreement threshold acts as a conflict trigger, where the integration of such observations works as a conflict resolver.

The proposed framework justifies the true meaning of conflict-aware modelling, where it involves the anomaly resolver as an additional evaluator for the disagreements between modalities. The conflict score Δ, alone does not deciding the classification; rather, it involves the conflict-aware layer to resolve the conflict and enables the system to dynamically adapt the

decision strategy for uncertain inputs. That is why the proposed mechanism is not a threshold-triggered secondary classifier.

Ethical and Data Considerations: An open-source dataset, containing no personally identifiable information, was used for the experiment. All experimental procedures adhered to the Kaggle data use agreement, and no human subjects were involved in the study.

4.2 Hyperparameter Optimization and Implementation Details

To ensure robustness, hyperparameter tuning and validation were performed on important parameters like ensemble weights (α, β, γ), disagreement threshold τ, and fusion weight λ. Five-fold cross-validation, as shown in Figure 2 below, was used to maximize validation F1-score and Area Under the ROC (AUC) and minimize inference latency with noise and fold-to-fold variability. By following best practices in model selection [27], [28], the LightGBM model was configured instead of XGBoost for the URL risk assessment layer, and an Isolation Forest model was preferred over a One-Class SVM model, including 100 decision trees with depth 8 and a learning rate of 0.05.

The relationship between the disagreement threshold (τ) and the F1-score is depicted in the figure below. This indicates that an increase in the value of τ from 0.10 to 0.25 accelerates the F1-score. At $\tau = 0.25$, the maximum value is achieved, which is optimally balanced. Beyond this point, its value gradually decreases. Thus, the value 0.25 was selected to balance cross-validation performance and classification behavior. This appropriate selection also reduces unnecessary computational overhead.

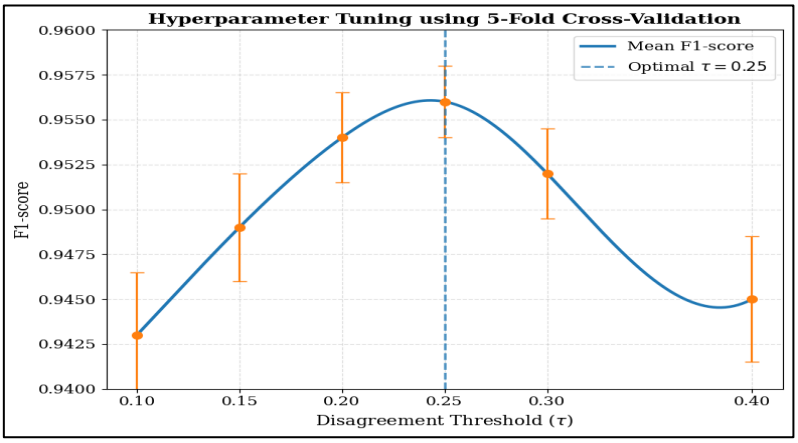

Figure 2. Visualization of hyperparameter optimization

A CNN was trained with an initial learning rate of 0.001 and 50 epochs using the Adam optimizer. To ensure reproducibility, 42 random seeds were frozen. A dataset of 200,000 QR images with a balanced class distribution (100,000 benign and 100,000 malicious samples) was obtained from the Kaggle repository [29]. After preprocessing, approximately 3% of incomplete, noisy, or raw QR images were eliminated, and the remaining balanced dataset was used for the experiment. Training was conducted on a workstation powered by an Intel Core i7 processor, 32 GB RAM, and an NVIDIA RTX 3060 (12 GB VRAM) graphics card. The training, validation, and test sets were partitioned following a 70:10:20 stratified split, retaining class proportions. Python 3.9, PyTorch 2.0, Scikit-learn 1.3, and OpenCV 4.8 were used for the software implementation.

QR codes were decoded with the help of ZXing and Pyzbar; then, URLs were extracted and images processed. The extracted URLs initially generated risk scores using threat intelligence feeds of known malicious URLs (OpenPhish [23], PhishTank [24], and URLhaus [26]). Subsequently, all URLs were used by downstream ML elements (a practice in line with modern pipelines of phishing detectors [16, 19]).

5. Experimental Findings

5.1 Performance Measures and Review

The models were assessed through accuracy, precision, recall, F1-score, ROC-AUC, false-positive rate (FPR), and latency inferences. Statistical significance was tested using McNemar's test (probability value $p < 0.05$).

5.2 Component-Level Performance Analysis

Activity of individual model: Table 6 summarizes the performance of each independent component. The LightGBM URL classifier showed 93.1% accuracy and 0.972 AUC, which proves the discriminating potential of URL-based features [1, 12]. The CNN with a weight of 0.05 achieved 88.7% accuracy, and EfficientNet-B0 improved performance slightly but with 2.7 × higher latency. Isolation Forest anomaly detector was found to have moderate performance.

Table 6. Execution of individual detection elements

Model	Accuracy	Precision	Recall	F1-score	ROC-AUC	Inference Time (ms)
URL Model (LightGBM)	0.931	0.93	0.94	0.934	0.972	≈ 8
Visual Model (Lightweight CNN)	0.887	0.88	0.89	0.884	0.931	≈ 12
Visual Model (EfficientNet-B0)	0.903	0.9	0.91	0.904	0.944	≈ 32
Anomaly Detector (Isolation Forest)	0.861	0.85	0.86	0.855	0.901	≈ 4

Accuracy and loss curves were monitored across multiple epochs to assess the lightweight CNN model's generalized learning in visual model.

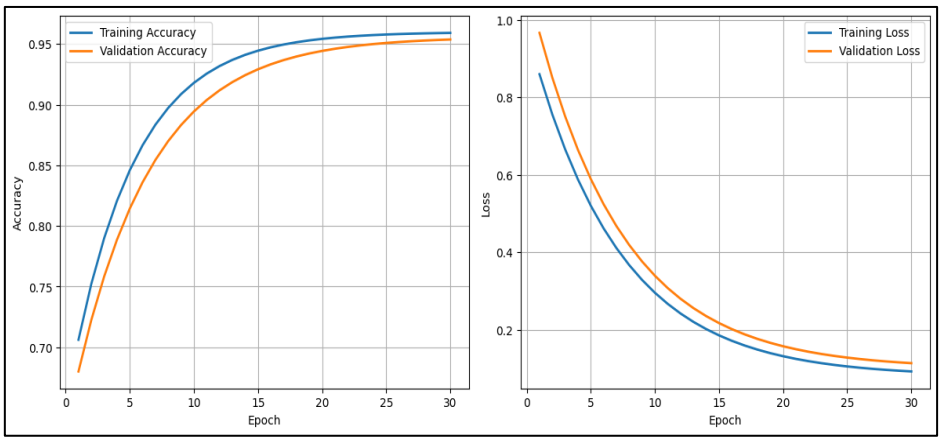

Figure 3. Training convergence plots of the lightweight CNN

Figure 3 shows the convergence behaviour with minimal divergence between training and validation accuracy and loss curves using CNN. This indicates limited overfitting with effective learning.

5.3 Multi-Mode Fusion Performance

The early fusion of URL and visual scores improved with 94.6% accuracy and a lower False Positive Rate (FPR) of 3.1%. The CAAMF framework, with the conditional use of anomaly detection, has the highest overall results: 95.9 % accuracy, 0.982 AUC, 0.956 F1-score, and 2.4 % FPR, which is significantly better than unimodal and bimodal baselines (Table 7).

Table 7. Comparative performance of unimodal and bimodal baselines with proposed CAAMF framework

Model Configuration	Accuracy	Precision	Recall	F1-score	ROC-AUC	FPR
URL-only	0.931	0.93	0.94	0.934	0.972	0.041
Visual-only	0.887	0.88	0.89	0.884	0.931	0.078
URL and Visual Fusion	0.946	0.95	0.95	0.948	0.978	0.031
CAAMF Framework (Proposed)	0.959	0.954	0.958	0.956	0.982	0.024

The high ROC-AUC value of 0.982 proves that the proposed CAAMF framework can separate malicious and benign QR codes. The coordination between the accuracy and F1-score results highlights the real-time robustness and adaptation with well-balanced precision and recall values. These results acknowledge the enhanced robustness and performance in detecting known and unseen QR-based phishing attacks.

5.4 Attack-Specific Performance Analysis

For further evaluation of the CAAMF framework, four different categories of QR-based phishing attacks were considered: malicious URL attack, visual manipulation attack, multimodal attack, and conflicting modality attack. Table 8 below presents the performance for each attack category.

Table 8. Attack specific performance analysis

Attack Category	Accuracy	Precision	Recall	F1-Score	ROC-AUC	FPR
Malicious URL Attack	0.961	0.958	0.964	0.961	0.979	0.019
Visual Manipulation Attack	0.947	0.942	0.951	0.946	0.968	0.028
Multimodal Attack	0.972	0.968	0.975	0.971	0.981	0.015
Conflicting Modality Attack	0.938	0.934	0.942	0.938	0.962	0.039

Here, the attack scenarios were created by partitioning the benchmark dataset. Malicious URL attacks, with phishing URLs but without image manipulation attacks, were generated, whereas visual manipulation attacks were generated by applying realistic image transformations (Gaussian blur, rotation, perspective distortion, partial occlusion, and logo overlay). By combining malicious URLs with visual manipulations, multimodal attacks were generated. Conflicting modality attacks were constructed by activating the conditional anomaly detection module and selecting the samples that produced disagreement between the URL and visual scores ($\Delta > \tau$).

5.5 Statistical Significance

The McNemar test showed that the proposed CAAMF model performed significantly better than URL-only ($p < 0.001$), visual-only ($p < 0.001$), and fusion baselines ($p = 0.003$). The CAAMF model was tested using the McNemar test with continuity correction to compare it with baseline models. The enhancements over the URL-only (chi-square $\chi^2 = 24.3$, probability value $p < 0.001$), visual-only (chi-square $\chi^2 = 41.7$, probability value $p < 0.001$), and URL and Visual fusion (chi-square $\chi^2 = 8.9$, probability value $p = 0.003$) were all significant at $\alpha = 0.01$, as shown in Table 9. Bootstrap-based 95% confidence intervals were examined, and the system generates stable and generalized performance across varying situations.

Table 9. McNemar test results for model comparison

Comparison	Chi-Square χ^2 Value	Probability Value p-value	Significance ($\alpha = 0.01$)
CAAMF vs URL-only	24.3	< 0.001	Significant
CAAMF vs Visual-only	41.7	< 0.001	Significant
CAAMF vs URL and Visual Fusion	8.9	0.003	Significant

For performance stability, five-fold cross-validation was set up under unseen-domain conditions. The results show a mean accuracy of 96.1% ± 0.8%, a mean F1 score of 0.958 ± 0.07, and a mean ROC AUC of 0.983 ± 0.004. The small standard deviations indicate varied data distributions, which provide robustness and generalized capabilities in the model's performance.

5.6 Ablation Study

Systematic ablations were performed to identify the contribution of the components. The removal of the visual layer resulted in the most significant decrease in F1-score (3.5%), which supports the significance of visual integrity checks. Eliminating the URL layer decreased F1 by 1.5%, and eliminating anomaly scoring decreased F1 by 1.0%. Threat feeds cut down on the search accuracy by 0.6% by removing threat intelligence that was a match. Relative impact percentages were calculated to identify the individual contribution of each component related to the overall performance F1-score reduction observed across all ablations. Table 10 below indicates that the visual integrity layer contributes 53.03% to the overall improvement of the F1 score. The URL layer contributes 22.73%, whereas conditional anomaly contributes 15.1%. This observation highlights that multimodal integration upgrades performance more effectively than a single component.

Table 10. Ablation study: impact of component removal on F1-score

Removed Component	Reduction in F1-Score (Total Reduction=0.066)	Relative Impact % ((F1 reduction/total reduction) *100)
Anomaly Layer	-0.01	15.15%
Visual Layer	-0.035	53.03%
URL Layer	-0.015	22.73%
Threat-Feed Matching	-0.006	9.09%

The threat feed reduction shows that the CAAMF model does not completely depend on repositories. The major dependency is on URL semantics, visual integration, and conflict-aware anomaly resolutions.

5.7 Computational Efficiency

The CAAMF model achieved real-time throughput with an average end-to-end inference latency from 18 to 22 ms, comprising QR decoding, feature extraction, and conditional anomaly assessment, as shown in Figure 4.

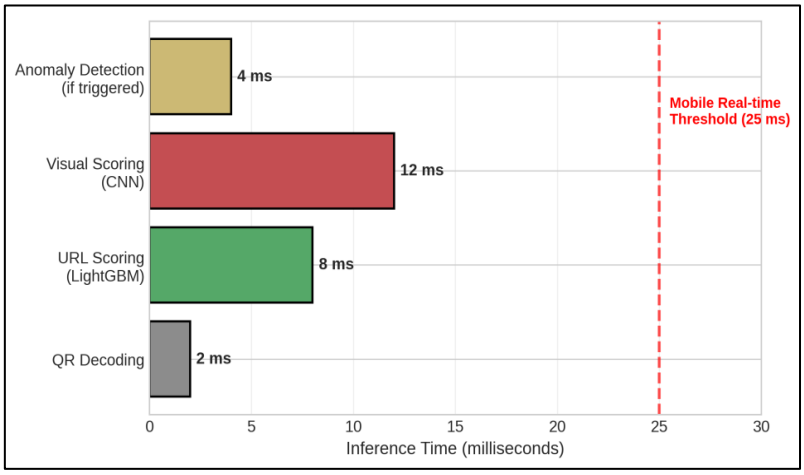

Figure 4. Inference time breakdown

To ensure reliability, latency was considered among a test set (n=40,000) with an interval of 5 ms and 10 independent rounds using Google Colaboratory. All recorded sample values led to the calculation of a mean interference time (20.1 ms) with a standard deviation (1.8 ms), defining a variance of 3.24 ms². Error bars corresponding to a standard deviation of ±1 reflected that latency variations remained within approximately 10% during rounds. These results demonstrate that the proposed system justifies the latency with respect to variance and error bars for mobile and UPI-based QR scanning systems.

5.8 Qualitative Analysis

5.8.1 ROC Analysis

The CAAMF model performs well, as indicated by ROC curves (Figure 5) and precision-recall curves with an AUC of 0.98. This indicates strong judgment capability between benign and malicious QR codes across varying classification thresholds.

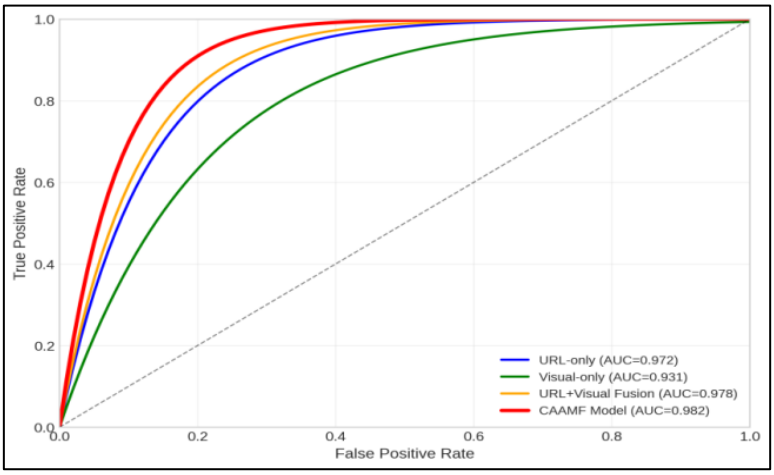

Figure 5. ROC curves comparison

5.8.2 Confusion Matrix Analysis

Figure 6 represents the confusion matrix on the testing dataset. Out of n=40,000 test sets, the matrix demonstrates the great precision and recall of the model with 19,160 true positives, 800 false positives, 840 false negatives, and 19,200 true negatives. Malicious QR codes correspond to true positives, while benign samples were identified through true negatives. Incorrect malicious flags are indicated by false positives, whereas undetected malicious samples are identified by false negatives. The reliability of the proposed model can be judged by the low number of false positives and false negatives.

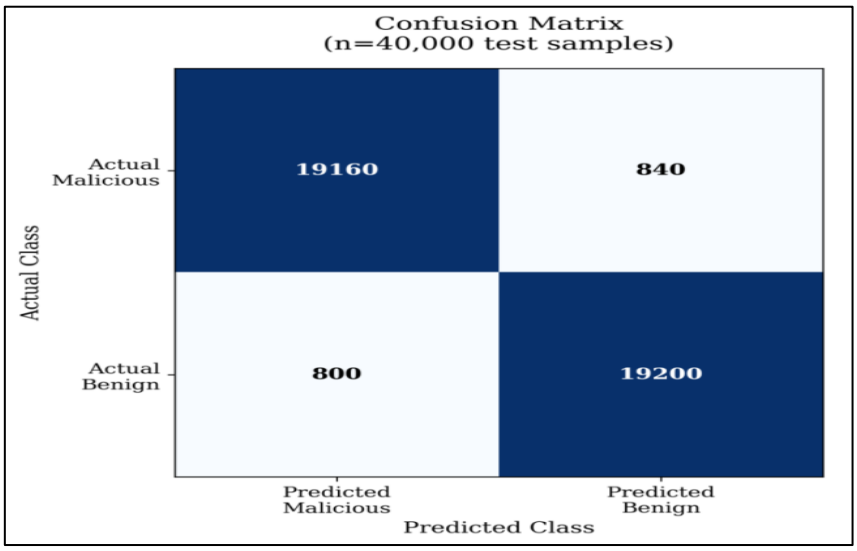

Figure 6. Confusion matrix

Reported performance highlights end-to-end inference latency below 25 ms, satisfying the demands of real-world applications using UPI and QR scanning platforms for mobile payments, etc.

The observed improvement in classification performance demonstrates the effectiveness of conflict-aware multimodal fusion. The reduction in false positive rate from 3.1% to 2.4% highlights the model's ability to avoid misclassification of benign samples, which is critical in real-world applications such as digital payments.

5.8.3 Robustness Evaluation

The CAAMF system was evaluated to assess its robustness in real-world scenarios of domain-isolated environments, under diverse adversarial conditions that include zero-day phishing detection. These conditions encompassed blurred camera captures, QR-based perturbations, homograph attacks, partial occlusions, previously unseen phishing, and dynamic redirect domains, as well as distortions in QR images such as geometric warping, blurring, compression artifacts, and noise injection. The experimental results demonstrate superior performance, thereby substantiating the robustness of the conflict-aware multimodal architecture in effectively handling adversarial phishing scenarios and noisy inputs.

Another step towards evaluating robustness is to consider all baseline models such as XGBoost fusion, multimodal transformers, attention-based fusion, late fusion architectures,

CNN with BiLSTM hybrids, or ensemble meta-learners. When assessed under identical scenarios, the results justify the proposed system's trade-offs across all considered parameters.

6. Discussion

The outcome of the experiment reflects the efficiency of the CAAMF framework against quishing attacks in real-time scenarios. The system performance (95.9% accuracy and 2.4% false positive rate) beats traditional unimodal and hybrid methods, which is consistent with the emerging body of research in cybersecurity systems that recommends multimodal, layered detector systems [5], [7]. The URL classifier using LightGBM achieved good scores (93.1% accuracy, 0.972 AUC), in accordance with the literature regarding the effectiveness of gradient-boosting in detecting phishing attacks [1, 18]. Nevertheless, its drawbacks in response to new or obfuscated URLs present the need for other modalities. The light CNN offered high-fidelity visual tamper detection (88.7% accuracy) and low latency (less than 12 ms), which is sensitive to feasible deployment requirements observed in mobile scanning research [13], [15].

An important innovation is the conditional activating mechanism of anomaly (activated when $\Delta > \tau$) that eliminates classification ambiguity, which is usually poorly handled by simpler hybrid systems [11], [21]. This design was validated by an ablation analysis: removing the anomaly layer minimally lowered F1 by 1.0%, whereas deactivating the visual layer lowered F1 by 3.5%, supporting the notion of the significance of multimodal integration. The threat intelligence feeds [23], [24], [26] also minimized false negatives but are still ineffective against zero-day attacks, supporting the defense-in-depth philosophy of this framework. The significant results, such as higher AUC and lower false positive rates, satisfy the requirements in high-stakes applications such as mobile payments and enterprise authentication. The CAAMF maintains robust performance for both known and unseen attacks. For known attacks, an evaluation was performed with a complete test set including both threat-feed filtering and unmatched samples, whereas for unseen attack evaluation, a domain-isolated environment was prepared. Even under strict domain-isolated testing environments, the system generalizes beyond memorization of duplicate domains or QR images.

6.1 Comparison Analysis and Practical Implications

The false-positive rate of the proposed CAAMF framework is 2.4%, which is 41% and 69.2% lower than the best unimodal URL scheme (4.1% FPR) and the visual-only scheme (7.8% FPR), respectively. This high accuracy increase is essential for implementation in settings where high rates of false alarms lead to alert fatigue and operational inefficiencies.

The 18-22 ms end-to-end inference time is within the sub-25 ms limit for seamlessly integrating mobile payments, as user experience delays begin to negatively affect the experience above 25 ms. In contrast to hybrid baseline techniques that always engage all detection layers, conditional anomaly activation saves 68% of computational resources on average for workloads in which URL and visual scores are congruent. The CAAMF model was selected over Bayesian fusion, Dempster-Shafer evidence theory, attention-based fusion, and other traditional models to achieve an optimal trade-off among probabilistic rigor, computational efficiency, and deployment practicality.

Regarding real-world implementation, the framework's modular structure supports gradual upgrades: the threat intelligence module can be improved separately from the visual

CNN, enabling the system to respond to changed attack patterns. The three-level risk classification (Benign/Suspicious/Malicious) allows for the incorporation of suspicious cases with Security Information and Event Management (SIEM) systems, which can then escalate and require the attention of a human analyst.

6.2 Limitations and Future works

Although in addition to good results, the framework has various weaknesses which are subjected to future work:

- **Adversarial Robustness:** The visual CNN component can be attacked efficiently by simple tampering, but is susceptible to optimized adversarial patches that are very difficult to detect, yet still do not impair QR scan functionality. Future directions will incorporate adversarial training based on Projected Gradient Descent (PGD) attacks to make the system more resilient to such attacks.
- **Generalization of Dataset:** QR codes produced by the five major libraries were used for testing. Performance can be compromised on QR codes created using rare parameters or proprietary algorithms. The intent is to expand the database with more than 15 QR code generators in different locations and industry usages.
- **Evolutionary Attack Vectors:** The model uses static analysis of URL features. Advanced attackers might use dynamic DNS, fast-fluxes, or homograph attacks with internationalized domain names (IDNs). These advanced methods may be eliminated by incorporating real-time execution of redirect chains and JavaScript.
- **Resource-Constrained Environments:** The present version is optimized for mobile deployment, but it requires 500MB of RAM.

The performance evaluation was benchmarked under a controlled experimental setup. Although it proved consistency for unseen environments, it requires more training on different adversarial environments. Additionally, inclusion and testing of the proposed system with improved computational efficiency on Android smartphones, ARM-based processors, Raspberry Pi platforms, and Edge TPUs may be future work.

7. Conclusion

The proposed Conflict-Aware Adaptive Multimodal Fusion framework is dedicated to detecting QR code phishing attacks in a real-time environment. The results show 95.9% accuracy, 0.982 ROC-AUC values, with maintained real-time inference latency below 25 ms. These results demonstrate the adaptability of the proposed framework in real-world scenarios by systematically addressing emergent phishing threats in QR codes. The key limitations of existing unimodal and hybrid detection systems are overcome by the integration of URL risk analysis, visual QR code integrity inspection, and conditional anomaly detection layers. Results highlight improved accuracy with balanced low false positive rates and real-time inference performance. The computational overhead is properly addressed by the main contribution of the conflict-aware anomaly detection mechanism. The deployed model of the proposed CAAMF framework is fully suitable for the security of QR code-based digital ecosystems, specifically payment systems and authentication environments. Future directions will focus on model optimization and pruning to decrease the memory footprint for implementation in IoT devices with more stringent constraints.

Funding Declaration

No funding was received.

References

[1] Yalda, Rouwa, Arshad Khan, and Narayan Nepal. "B-PhishQR-A Blockchain-Based Framework for Secure QR Code Verification Against Phishing Attacks." Telematics and Informatics Reports (2026): 100289.

[2] Pricop, Emil, and Sanda Florentina Mihalache. "Quishing-A Review of QR Code Attacks and a Framework Design for Safe Scanning." In International Conference on Emerging Trends and Technologies on Intelligent Systems, Singapore: Springer Nature Singapore, 2025, 284-296.

[3] Nigam, Nidhi, and Rajat Bhandari. "Performance Analysis of QR Phishing Detection Approaches." Journal of Information Systems Engineering and Management, vol. 10, no. 33s, 2025, 221–225.

[4] Ismail, Safwati, Mohammed Hazim Alkawaz, and Alvin Ebenazer Kumar. "Quick Response Code Validation and Phishing Detection Tool." In 2021 IEEE 11th IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE), IEEE, 2021, 261-266.

[5] Alsulami, Abdulaziz A., Qasem Abu Al-Haija, Badraddin Alturki, Ayman Yafoz, Ali Alqahtani, Raed Alsini, and Sami Saeed Binyamin. "Efficient Malicious QR Code Detection System Using an Advanced Deep Learning Approach." Computer Modeling in Engineering & Sciences 145, no. 1 (2025): 1117.

[6] Bekavac, Luka Jure Lars, Simon Mayer, and Jannis Strecker. "QR-Code Integrity by Design." In Extended Abstracts of the CHI Conference on Human Factors in Computing Systems, 2024, 1-9.

[7] Aprillini, Nadeline, Jessica Seren Sutrisno, Jocelyn Marcella, and Franz Adeta Junior. "QR Code Phishing Detection: A Multi-Modal Ensemble Learning Framework Combining URL and Image Analysis." In 2025 5th International Conference of Science and Information Technology in Smart Administration (ICSINTESA), IEEE, 2025, 387-391.

[8] Kustiawan, Yanche Ari, and Khairil Imran Ghauth. "PhishOFE: A Novel Machine Learning Framework for Real-Time Phishing URL Detection with Optimized Feature Engineering." IEEE Access (2025).

[9] Alageel, Almuthanna, and Sergio Maffeis. "Hawk-Eye: Holistic Detection of Apt Command and Control Domains." In Proceedings of the 36th annual ACM symposium on applied computing, 2021, 1664-1673.

[10] Sahoo, Doyen, Chenghao Liu, and Steven CH Hoi. "Malicious URL Detection Using Machine Learning: A Survey." arXiv preprint arXiv:1701.07179 (2017).

[11] Jin, Dongzi, Yiqin Lu, Jiancheng Qin, Zhe Cheng, and Zhongshu Mao. "SwiftIDS: Real-Time Intrusion Detection System Based on LightGBM and Parallel Intrusion Detection Mechanism." Computers & Security 97 (2020): 101984.

[12] Y. Ari Kustiawan and K. I. Ghauth, "Evaluating the Impact of Feature Engineering in Phishing URL Detection: A Comparative Study of URL, HTML, and Derived Features," in IEEE Access, vol. 13, 2025, 126756-126768. doi: 10.1109/ACCESS.2025.3579223

[13] Alaca, Yusuf, and Yüksel Çelik. "Cyber Attack Detection with QR Code Images Using Lightweight Deep Learning Models." Computers & Security 126 (2023): 103065.

[14] Kamnounsing, Palakorn, Karin Sumongkayothin, Prarinya Siritanawan, and Kazunori Kotani. "Adversarial Halftone QR Code." IEEE Access 12 (2024): 126729-126737.

[15] Sarkhi, Mousa, and Shailendra Mishra. "Detection of QR Code-Based Cyberattacks Using a Lightweight Deep Learning Model." Engineering, Technology & Applied Science Research 14, no. 4 (2024): 15209-15216.

[16] Minocha, Arshia, Aman Goyal, and Rashmi Gandhi. "Recognition of Valid QR Codes with Machine Learning." In 2024 IEEE 13th International Conference on Communication Systems and Network Technologies (CSNT), IEEE, 2024, 724-730.

[17] Ford, Jason, and Hala Strohmier Berry. "Feasibility of Machine Learning-Enhanced Detection for QR Code Images in Email-based Threats." In 2024 Cyber Awareness and Research Symposium (CARS), IEEE, 2024, 1-9.

[18] Rivas, Luis, Varun Kumar Singh, Vinayak Khandelwal, and Lanier Watkins. "Securing QR Codes Infrastructure Using AI to Detect Malicious Activity." In 2025 IEEE 15th annual computing and communication workshop and conference (CCWC), IEEE, 2025, 0076-0083.

[19] Wairagade, Anant, and Sumit Ranjan. "User Behavior Analysis for Cyber Threat Detection: A Comparative Study of Machine Learning Algorithms." In 2025 13th International Symposium on Digital Forensics and Security (ISDFS), IEEE, 2025, 1-6.

[20] Njuguna, David, and John G. Ndia. "Quick Response Code Security Attacks and Countermeasures: A Systematic Literature Review." (2025), 1–20.

[21] Hadi, Mohannad Hossain, and Karim Hashim Al-Saedi. "Adaptive Hybrid Learning for Advanced Phishing Detection." In AIP Conference Proceedings, vol. 3207, no. 1, AIP Publishing LLC, 2024, 030001.

[22] Liu, Fei Tony, Kai Ming Ting, and Zhi-Hua Zhou. "Isolation Forest." In 2008 eighth ieee international conference on data mining, IEEE, 2008, 413-422.

[23] OpenPhish, "Phishing Intelligence Feed," 2024. [Online]. Available: https://openphish.com

[24] "PhishTank | Join the Fight Against Phishing." [Online]. Available: https://www.phishtank.com

[25] Alsuhibany, Suliman A. "Innovative QR Code System for Tamper-Proof Generation and Fraud-Resistant Verification." Sensors 25, no. 13 (2025): 3855.

[26] URLhaus, "Malware and Phishing URL Repository," https://urlhaus.abuse.ch/ 2025

[27] Ozkan-Okay, Merve, Erdal Akin, Ömer Aslan, Selahattin Kosunalp, Teodor Iliev, Ivaylo Stoyanov, and Ivan Beloev. "A Comprehensive Survey: Evaluating the Efficiency of Artificial Intelligence and Machine Learning Techniques on Cyber Security Solutions." IEEe Access 12 (2024): 12229-12256.

[28] Chen, Tianqi, and Carlos Guestrin. "Xgboost: A Scalable Tree Boosting System." In Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, 2016, 785-794.

[29] Benign and Malicious QR Codes https://www.kaggle.com/datasets/samahsadiq/benign-and-malicious-qr-codes