International Journal of Safety and Security Engineering
Vol. 16, No. 2, February, 2026, pp. 275-282
Journal homepage: http://iieta.org/journals/ijsse

A Deep Learning Framework for Covert Timing Channel Detection Using Gramian Angular Field and Markov Transition Field Fusion and Convolutional Neural Networks

Shorouq Al-Eidi

Computer Science Department, Tafila Technical University, Tafila 66110, Jordan

Corresponding Author Email: saleidi@ttu.edu.jo

https://doi.org/10.18280/ijsse.160203

Received: 10 December 2025
Revised: 30 January 2026
Accepted: 20 February 2026
Available online: 28 February 2026

Keywords:
Covert Timing Channel detection, time-series to image encoding, Gramian Angular Field, Markov Transition Field, Convolutional Neural Networks, network intrusion detection, deep learning framework

ABSTRACT

Covert Timing Channels (CTCs) represent a significant security threat in network communications, they rely on the use of inter-arrival times in embedding secret information without being detected using conventional mechanisms. This study introduces a deep learning framework that integrates the Gramian Angular Field (GAF) and Markov Transition Field (MTF) to enhance CTC detection. By transforming one-dimensional packet timing data into two-dimensional image representations, the framework leverages Convolutional Neural Networks (CNNs) to capture complex temporal and state-dependent patterns. Evaluated on a dataset of 6,000 network traffic sequences (3,000 benign and 3,000 covert), experimental results show that the GAF-MTF fusion model outperforms traditional machine learning methods with an accuracy of 96.8%, precision of 97.1%, recall of 96%, and F1-score of 96.5%. This demonstrates the effectiveness of combining time-series image transformation techniques with deep learning to detect covert communications, offering a scalable solution for enhancing network intrusion detection systems.

1. INTRODUCTION

In the current digital world, networks are vulnerable to various forms of attacks which use behavioral characteristics of communication protocols. The category of such attacks includes the Covert Timing Channels (CTCs) where data is encoded based on packet timing and not the content itself, making them difficult to detect using conventional security tools.

Conventional methods used for detecting the CTC attacks like statistical methods, entropy, and rule based methods use manual feature creation and prior knowledge about the normal traffic flows. Even though these methods are able to detect simple patterns in traffic, they fail to incorporate the intricate temporal relationships found within current network flows, particularly under circumstances where there is significant noise in the flow data. Machine learning algorithms have been effective in improving the efficiency of CTC detection through learning traffic patterns, however, they also still depend on handcrafted features.

Recent advances in deep learning have opened new opportunities for analyzing network traffic through automatic learning of complex feature representation. In particular, transforming time series data into image-based representations has emerged as an effective strategy for leveraging the powerful feature extraction capabilities of Convolutional Neural Networks (CNNs). Techniques such as Gramian Angular Field (GAF) and Markov Transition Field (MTF) enable the conversion of one-dimensional temporal sequences into two-dimensional images while preserving important temporal characteristics. GAF represents global temporal correlation and magnitude information, while MTF is concerned with the transition probability between the states of the time series.

Despite their individual effectiveness, many of the existing methods make use of only one type of representation, which may limit their ability to fully capture the complexity of covert timing patterns. To address this limitation, this paper proposes a new deep learning architecture that takes advantage of both GAF and MTF representations to create a multi-channel input for CNN-based classification. With the combined representations, the proposed method is expected to be capable of learning more complex spatio-temporal information. The proposed framework is evaluated using a dataset of network traffic flows containing both benign and covert communications. Experimental results demonstrate that the fusion of GAF and MTF representations is much more effective than any single representation or conventional machine learning algorithms. In addition, the statistical test proves that the obtained results are reliable and significant.

The remainder of this paper is organized as follows: Section 2 reviews related work on CTCs and time series imaging techniques. Section 3 describes the proposed methodology, regarding data encoding and model architecture. Section 4 presents the experimental results and performance analysis. Finally, Section 5 concludes the paper and outlines future research directions.

2. RELATED WORK

This section reviews literature related to the detection of CTCs and network traffic analysis. The review includes classical statistical methods, machine learning models, deep learning techniques, and innovative time series imaging techniques.

2.1 Covert Timing Channel detection

CTCs are considered a significant threat to network security because they permit secret communication by using the timing of packets rather than their contents. This makes it hard to detect such channels using conventional intrusion detection mechanisms. Previous studies on the detection of CTCs were based on the statistical analysis of packet inter-arrival times [1]. In their study, Han et al. [2] proposed a mechanism for detecting CTCs using the analysis of packet timing intervals and payload characteristics. In another study, Li et al. [3] proposed an anomaly detection mechanism for CTCs, which has the ability to detect such channels through the modeling of normal traffic timing patterns.

However, subsequent research focused on applying various techniques of machine learning to detect covert channels. Machine learning techniques analyze the traffic patterns in a network and then classify them to detect any kind of covert channel. Yazykova et al. [4] in their paper discussed various machine learning techniques to detect CTC under various encoding schemes and traffic conditions. Their results showed that machine learning techniques offer improved flexibility compared to traditional statistical techniques. Elsadig and Gafar [5] also provided a detailed survey that showed the effectiveness of various machine learning techniques, such as Support Vector Machines and Random Forests, to detect covert channels. Moreover, previous work [6, 7] has presented an image-based method for the detection of CTCs. According to their method, the sequence of packet inter-arrival times is converted into an image format and fed into a CNN.

Recent research also focused on applying various techniques of deep learning to detect covert channels. Deep learning techniques are also capable of learning from data and then classifying it to detect any kind of covert channel. For instance, Al-Eidi et al. [8] proposed a deep learning technique that analyzes the sequential inter-arrival times of packets to detect any CTC, achieving strong performance without relying on handcrafted statistical features. Moreover, Sun et al. [9] presented a deep learning method for the detection of CTC based on the Auxiliary Classifier Generative Adversarial Network (ACGAN). In their work, the sequence of packet inter-arrival times is converted into images using the GAF, and the GAN is used to learn discriminative features for the detection of CTC. From the experiment results, high accuracy and robustness are achieved for different CTC types. Recently, Darwish et al. [10] proposed a framework called LinguTimeX, which focuses on the detection of CTCs using NLP and XAI techniques. It treats network traffic as a linguistic sequence and detects abnormal timing behaviors in various communication environments. The model explainability will make it easier to interpret the findings of the analysis, thus demonstrating the efficiency of applying NLP-based feature extraction in combination with XAI.

2.2 Deep learning for network traffic analysis

The deep learning approach has gained significant attention lately for its applications in network security because of its capability of extracting highly complicated information from massive data. The use of deep learning in different network security applications has become widespread, including intrusion detection, traffic classification, and anomaly detection.

Javaid et al. [11] proved that it was possible to successfully use deep neural networks to detect malicious actions performed in the network. Another work of relevance was performed by Wang et al. [12]. These researchers used a CNN-based solution that could successfully categorize encrypted traffic even if there was no information about its payload. Moreover, GANs have also been used in the field of cybersecurity research. Furthermore, the researchers Meidan et al. [13] suggested a machine learning-based solution, which could successfully identify unauthorized IoT devices through traffic pattern analysis. Anomalies are detected through behavioral features extraction and supervised learning, which helps distinguish the legitimate devices from potentially malicious ones. There are several benchmark datasets available that can be used to evaluate the effectiveness of IDS. One of the datasets is the UNSW NB15 dataset proposed by Moustafa and Slai [14], and the CICIDS dataset proposed by Sharafaldin et al. [15], which contain realistic network traffic and labeled attack scenarios that can be employed for training and testing various ML and DL models.

Goodfellow et al. [16] presented GANs as a generative model composed of a generator network and a discriminator network, trained in an adversarial fashion. In network security, GANs have been considered as an approach to enhance intrusion detection systems. Moreover, Lin et al. [17] have presented IDSGAN, which generates adversarial network traffic samples to evaluate intrusion detection systems. Moreover, Rigaki and Garcia [18] have shown the capability of GANs to adapt malware communication patterns to evade intrusion detection systems. Recently, Mari et al. [19] have presented a machine learning-based intrusion detection system (IDS). The authors have also shown the impact of Generative Adversarial Networks (GANs) on intrusion detection system performance. The authors have considered the NSL KDD dataset to train classical ML-based intrusion detection systems and have presented a GAN model to generate adversarial network traffic samples, which can evade intrusion detection systems.

2.3 Time series imaging techniques for network traffic

Another promising direction in network traffic analysis involves transforming time series data into image representations, enabling the application of computer vision techniques for classification. Wang and Oates [20, 21] introduced the concept of encoding time series signals as images using GAF and MTF. These methods convert one-dimensional sequences into two-dimensional matrices while preserving temporal dependencies.

The GAF represents time series values in polar coordinates and constructs a Gramian matrix that captures temporal correlations between data points. In contrast, the MTF models the transition probabilities between quantized states of the sequence, allowing dynamic temporal behavior to be represented spatially. Several studies have demonstrated the effectiveness of these representations for time series classification tasks. Vargas et al. [22] applied GAF and MTF transformations for ordinal time series classification and

showed that combining these representations improves classification performance. Costa et al. [23] proposed a framework that fuses multiple time series image representations, including GAF, MTF, and recurrence plots, enabling deep learning models to extract richer spatio-temporal features. Similarly, Yang et al. [24] demonstrated that transforming multivariate time series data into GAF and MTF images allows CNN models to achieve performance comparable to deeper neural networks. Wang et al. [25] further explored multi-CNN architectures for analyzing different temporal components of time series data, achieving improved predictive performance.

Although substantial progress has been made in the detection of covert channels and time series analysis, existing methods often rely on either statistical features or a single time series encoding scheme. Notably, various time series encoding methods focus on different aspects of time series behavior. For instance, GAF mainly focuses on temporal correlations, while MTF is based on state transition dynamics. With this understanding of existing methods and their limitations, this paper proposes a hybrid deep learning-based framework that incorporates both GAF and MTF representations with a CNN classifier.

3. METHODOLOGY

This section describes the proposed framework for detecting CTCs using a hybrid deep learning technique employing GAF and MTF. The framework consists of three main stages: data collecting and preprocessing, time series encoding, and classification. First, raw network traffic is processed to extract packet inter-arrival time (IAT) sequences, which are then normalized. In the following step, the normalized sequences are transformed two-dimensional image representations using the GAF and MTF approaches, where each channel is responsible for providing different time characteristics. Finally, the combined images are fed into a CNN as inputs to extract the distinguishing discriminative features and perform the final classification of traffic as benign or covert as illustrated in Figure 1.

3.1 Dataset overview

The dataset used in this study was collected from a real-world network environment to reflect realistic traffic conditions such as network jitter and congestion. Network traffic was captured using a Wireshark tool and a TCP connection for both benign and covert communications.

CTCs were created by embedding binary information within packet IATs. Specifically, controlled delays were introduced such that a delay of 20 ms represents a binary '1', while a delay of 10 ms or less represents a binary '0'. The dataset contains 6000 sequences of benign and covert traffic equally divided between benign and covert classes (3,000 each). Each sequence contains a series of IAT values with varying lengths (256, 1024, and 4096). The dataset is then split into training (60%), validation (20%), and testing (20%) sets to ensure reliable model evaluation as shown in Table 1.

Table 1. Dataset summary and experimental setup

Dataset Split	# Sequences	Class Distribution (Benign / Covert)	Sequence Length (IATs)
Training	3,600	1,800 / 1,800	256,1024,4096
Validation	1,200	600 / 600	256,1024,4096
Testing	1,200	600 / 600	256,1024,4096
Total	6,000	3,000 / 3,000	256,1024,4096

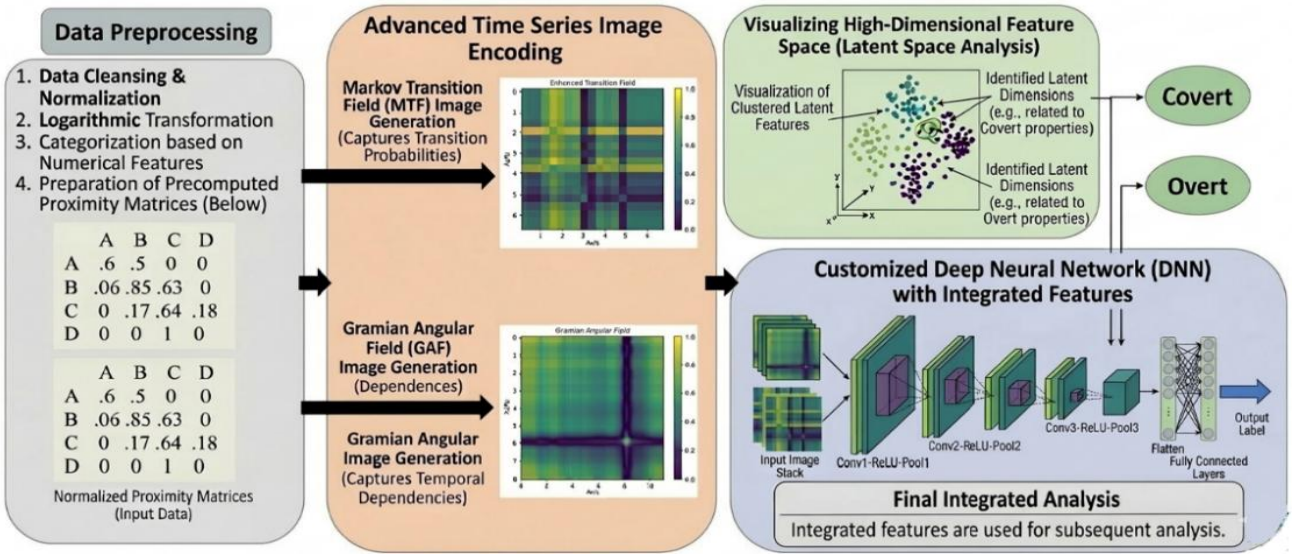

Figure 1. A deep learning framework for CTC detection using GAF-MTF fusion and CNNs
Note: Covert Timing Channel (CTC); Gramian Angular Field (GAF) and Markov Transition Field (MTF); Convolutional Neural Networks (CNNs).

3.2 Image-based encoding of inter-arrival times

Two encoding schemes were employed in this study to transform IAT sequences into images: GAF and MTF. GAF maps each time point to polar coordinates and builds a Gramian matrix in which each entry stores the cosine of the sum of angles between points. It thus preserves both magnitude and temporal relationships. MTF quantizes the sequence into quantile bins and calculates the probabilities of transitions among these states over time, capturing the

dynamic behavior of the sequence. Both methods transform the one-dimensional temporal IAT sequences to two-dimensional spatial representations, resized to 64×64 pixels to ensure uniform input for the CNN. Further details on GAF and MTF encoding processes are provided below.

3.2.1 Gramian Angular Field

The GAF is an encoding of temporal relationships in a time series by mapping the normalized values into polar coordinates. It inherently preserves temporal dependence and absolute temporal relationship information, which may otherwise be lost through many traditional Cartesian representations. For a time series of IATs $X = \{x_1, x_2,...,x_n\}$, where each x_i represents the time between consecutive events, the series is first rescaled to the range [-1, 1] using the formula below:

$$\tilde{x}_i = 2 \times \frac{x_i - \min(X)}{\max(X) - \min(X)} - 1 \quad (1)$$

The normalized series $\tilde{X}$ is then mapped to polar coordinates, with the IAT values determining the angular component and timestamps defining the radial component:

$$\begin{cases} \emptyset_i = \arccos(\tilde{x}_i), -1 < \tilde{x}_i \leq 1 \\ r_i = \dfrac{t_i}{N}, \qquad t_i \in \mathbb{N} \end{cases} \quad (2)$$

where,

- φ_i = angular component (IAT magnitude)
- r_i = radial component (normalized timestamp)
- N = normalization constant
- n = sequence length
- t_i is the timestamp of the i-th IAT

This polar representation has two important properties:

- **Bijectivity:** The mapping is invertible for $\phi \in [0, \pi]$, ensuring a unique polar representation for each time series.
- **Temporal Preservation:** Polar coordinates retain absolute temporal relationships, unlike Cartesian representations, where areas depend on intervals rather than absolute positions.

$$G = \begin{bmatrix} cos(\phi_1 + \phi_1) & \cdots & cos(\phi_1 + \phi_n) \\ cos(\phi_2 + \phi_1) & \cdots & cos(\phi_2 + \phi_n) \\ \vdots & \ddots & \vdots \\ cos(\phi_n + \phi_1) & \cdots & cos(\phi_n + \phi_n) \end{bmatrix}$$

Finally, the GAF is constructed as a Gramian matrix where each element G_{ij} = cos (sum of angles). This is an effective method for transforming IAT sequences to well-defined images based on the cosine summing of angles. An application of this can be the visualization and classification of timing patterns in network traffic.

3.2.2 Markov Transition Field

The MTF captures the dynamic evolution of IAT sequences by encoding the transition probabilities between quantized states over time. The IAT sequence $X = \{x_1, x_2,...,x_n\}$ is first quantized into Q quantile bins. Each data point xi is assigned to a bin qi∈ [1, Q]. A Q×Q Markov transition matrix W is then computed, where:

$$W_{ij} = P(x_{t+1} \in q_j \,|x_t \in q_j) \quad (3)$$

The MTF M is an n×n matrix obtained by mapping these probabilities along the temporal axis:

$$M = \begin{bmatrix} w_{ij|x_1 \in q_i, x_1 \in q_j} & \cdots & w_{ij|x_1 \in q_i, x_n \in q_j} \\ w_{ij|x_2 \in q_i, x_2 \in q_j} & \cdots & w_{ij|x_2 \in q_i, x_n \in q_j} \\ \vdots & \ddots & \vdots \\ w_{ij|x_n \in q_i, x_1 \in q_j} & \cdots & w_{ij|x_n \in q_i, x_n \in q_j} \end{bmatrix}$$

Key properties of MTF include:

- **Multi-span Transitions:** M_{ij} with |i-j|=k encodes transitions across time gaps of k.
- **Temporal Dynamics:** The diagonal M_{ii} retains self-transition probabilities.
- **Quasi-Gramian Structure:** Temporal order is preserved while storing transition statistics.

To reduce computational complexity, the MTF is typically down sampled via averaging over non-overlapping m×m patches:

$$m = \left\lceil \frac{n}{S_{MTF}} \right\rceil \quad (4)$$

where, S_{MTF} is the target image size.

3.2.3 Combined Gramian Angular Field and Markov Transition Field representation

The merged GAF-MTF representation embeds both the images of GAF and MTF into a single two-channel image, maintaining both the static temporal correlations and dynamic transition patterns. First, the IAT sequence is transformed into the GAF matrix $G \in R^{n \times n}$ that captures absolute temporal relations and morphological features, and the MTF matrix $M \in R^{n \times n}$ that encodes the probabilities of transition between quantized states over time. Both matrices are normalized to [0,1], in order to make them compatible for fusion. Then, along the channel dimension, the two matrices are stacked to form a fused image $F \in R^{n \times n \times 2}$, in which the first channel corresponds to the GAF image and the second channel corresponds to the MTF image. This two-channel representation can be fed into image-based models directly, such as CNNs, which allows the model to learn both the morphological features from GAF and dynamic state transitions from MTF. By fusing these complementary representations, the fused GAF-MTF image enhances feature richness and preserves effective spatio-temporal structure with improved classification performance compared to either method.

3.3 Convolutional Neural Network-based detection framework

In this work, the task of CTC detection has been carried out using a CNN based on GAF and MTF image representations of the IAT sequences.

The CNN architecture used has four convolutional layers with 32, 64, 128, and 256 filters, respectively. These are followed by a ReLU activation function, as well as 2×2 max-pooling. Three fully connected layers (128, 64, and output layer with softmax). To avoid overfitting, dropout has been used with a dropout rate of 0.5. The Adam optimizer with a learning rate of 0.001 has also been used. For GAF, an IAT sequence X of length n is transformed directly into an n×n Gramian matrix. For MTF, the sequence is quantized into Q bins to construct a Q×Q Markov transition matrix, which is

then down sampled to an S_{MTF}×SMTF matrix using an averaging kernel of size m × m, where m=⌈n/SMTF⌉.

The CNN is trained using image sizes $S_{MTF} \in \{16,32,64\}$ and quantization levels $Q \in \{16,32,64\}$, enabling effective learning of temporal correlations and transition dynamics for CTC detection.

3.4 Performance evaluation

The models were evaluated using a suite of standard classification metrics to assess their ability to detect CTC accurately. These metrics provide insights into various aspects of model performance and include the following:

$$\text{Accuracy} = \frac{(\text{TP} + \text{TN})}{(\text{TP} + \text{TN} + \text{FP} + \text{FN})} \quad (5)$$

$$\text{Precision} = \frac{\text{TP}}{(\text{TP} + \text{FP})} \quad (6)$$

$$\text{Recall} = \frac{\text{TP}}{(\text{TP} + \text{FN})} \quad (7)$$

$$\text{F1}-\text{score} = \frac{2 \times (\text{Precision} \times \text{Recall})}{(\text{Precision} + \text{Recal})} \quad (8)$$

These metrics were computed for both the overt and covert classes, and results were compared across standalone and hybrid representation models to determine the most effective detection strategy.

4. RESULTS AND ANALYSIS

This section presents the extensive performance evaluation of the proposed CTC detection framework using GAF, MTF, and fused GAF-MTF representations. The performances of all models were evaluated in terms of accuracy, precision, recall, F1-score, Receiver Operating Characteristic (ROC) analysis, error metrics, statistical significance testing, and the loss behavior during training and validation.

Table 2 reports the main classification results. Fused GAF-MTF has the best performance with an accuracy of 96.8%, which is accompanied by improvements in precision 97.1%, recall 96.0%), and F1-score 96.5%. This indicates that fusing GAF and MTF captures additional complementary patterns in the temporal and spatial dimensions of the encoded IAT sequences, thus leading to more discriminative features for CNN-based models.

Moreover, the proposed model was compared with several models as shown in Table 3. Classical approaches, including entropy-based detection, SVM, and Random Forest, attain 81.3-89.1% accuracy notably lower than the fused CNN. This demonstrates that deep spatio-temporal representation learning is crucial for CTC detection.

In order to further contextualize the performance of the proposed model, the fused CNN was compared with other deep learning architectures. All the models were tested using the same dataset splits and processing pipeline, wherein the IAT sequences were converted into image representations for fair comparison. It can be observed that the accuracy and F1-scores for the EfficientNet-B3 and ResNet50 models are slightly lower than the proposed model, while their computational complexity is significantly higher. This may be attributed to the fact that these models were originally intended for large-scale natural image classification tasks and may not perform optimally for the specific spatio-temporal patterns present in the GAF-MTF images.

Table 2. Performance of different image-based representations

Method	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
GAF-only	95.2	95.0	94.8	94.9
MTF-only	93.5	94.0	92.5	93.2
GAF-MTF	96.8	97.1	96.0	96.5

Note: GAF-MTF: Gramian Angular Field (GAF) and Markov Transition Field (MTF)

Table 3. Performance comparison of different models

Model	Accuracy (%)	Precision (%)	F1-Score (%)
Proposed model	96.8	97.1	96.5
Entropy-Based	81.3	79.6	80.8
SVM	87.4	86.8	86.6
Random Forest	89.1	88.2	88.5
EfficientNet-B3	96.1	96.5	96.1
ResNet50	95.8	95.9	95.7

According to the ROC curves in Figure 2, the suggested CNN with fusion of GAF-MTF shows better performance than others regarding detection of covert traffic flow compared with normal one. However, while the value of AUC for the Random Forest method has been (0.902), the AUC values for the other methods are not explicitly reported in the legend.

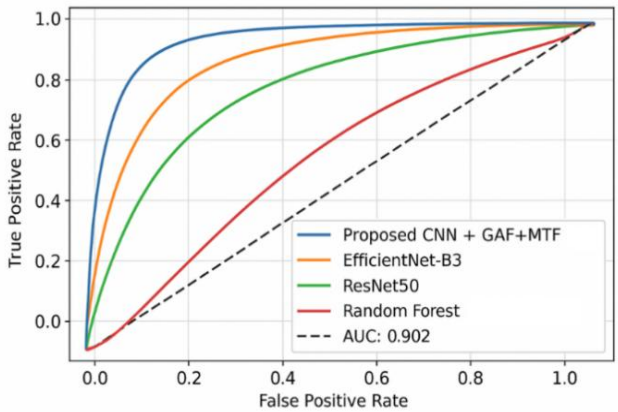

Figure 2. Receiver Operating Characteristic (ROC) curves for Convolutional Neural Network (CNN) architectures

Moreover, Table 4 shows that the combined GAF-MTF representation achieves the best performance, with the lowest error values including Mean Squared Error (MSE) = 0.021 and Mean Absolute Error (MAE) = 0.011 compared to GAF-only (0.034, 0.018) and MTF-only (0.041, 0.022). This indicates that integrating both representations improve accuracy, as they capture complementary temporal patterns better than either

method alone This indicates that integrating both representations improve accuracy, as they capture complementary temporal patterns.

Figure 3 depicts that the fused model converges faster and exhibits a smaller generalization gap, indicating effective learning and reduced overfitting during training.

To ensure that the model performance gains are not due to random variation, we conducted statistical significance tests using the independent test set. For this, the McNemar test has been performed at the sequence level, focusing on the cases where the two models have disagreed in their prediction results. The results have shown in Table 5 a statistically significant performance improvement of the GAF-MTF fused model over the performance of the two individual models. For the comparison between GAF and the fused model yielded the $\chi^2 = 11.42$ ($p < 0.001$). For the comparison between the MTF and the fused model, resulted in $\chi^2 = 19.83$ ($p < 0.0001$). Additionally, 95% confidence intervals were used to evaluate model accuracy.

Table 4. Error metrics of different image based representations

Representation	MSE	MAE
GAF-only	0.034	0.018
MTF-only	0.041	0.022
GAF-MTF	0.021	0.011

Note: Gramian Angular Field (GAF); Markov Transition Field (MTF); Mean Squared Error (MSE); Mean Absolute Error (MAE)

Table 5. Contingency tables for McNemar test

(a) GAF-MTF) vs. GAF		
	GAF Correct	**GAF Incorrect**
(GAF-MTF) Correct	560	120
(GAF-MTF) Incorrect	45	475
(b) (GAF-MTF) vs. MTF		
	MTF Correct	**MTF Incorrect**
(GAF-MTF) Correct	570	130
(GAF-MTF) Incorrect	25	475

Note: GAF-MTF: Gramian Angular Field (GAF) and Markov Transition Field (MTF)

Finally, the effect of image resolution was explored at 64 × 64, 32 × 32 and 16 × 16 inputs in Table 6. While performance degraded marginally with reduced resolution, the used model remains consistently superior and maintains strong accuracy of 64×64 resolution. This means it is robust and suitable for resource-constrained environments.

Table 6. Performance comparison across image sizes

Size	Method	Accuracy	Precision	Recall	F1-Score
64 × 64	GAF	95.9	95.0	96.5	95.7
	MTF	94.8	93.9	95.4	94.6
	GAF-MTF	96.8	96.3	96.0	96.9
32 × 32	GAF	94.7	93.8	95.1	94.5
	MTF	93.2	92.1	94.0	93.0
	GAF-MTF	95.8	95.2	96.4	95.7
16 × 16	GAF	92.8	91.9	93.4	92.6
	MTF	91.0	90.1	92.0	91.0
	GAF-MTF	93.9	93.2	94.5	93.8

Note: GAF-MTF: Gramian Angular Field (GAF) and Markov Transition Field (MTF)

The superior performance of the fused representation of the GAF-MTF can be explained in light of the complementary nature of these two coding schemes. Indeed, the GAF representation is able to encode global temporal correlations and magnitude relations in the IAT sequences, while the MTF representation is able to encode the probabilistic transitions between quantized states, reflecting dynamic temporal behaviors. Thus, the fused input representation provides a richer and more discriminative spatio-temporal feature space that allows the CNN to learn the feature representations in a more effective manner.

Despite the strong performance, the proposed method has some limitations. One of these limitations is related to misclassification, especially in situations where there is a similarity in timing patterns and normal traffic behavior, especially under noisy and highly variable network conditions. Moreover, there is also a limitation related to fixed-size IAT sequences and parameters, especially in highly dynamic and real-time situations. In addition, image-based models also introduce extra computational complexity during preprocessing stages. Furthermore, the model has been evaluated on a specific dataset, and its generalization to other network settings or unseen covert channel strategies requires further investigation.

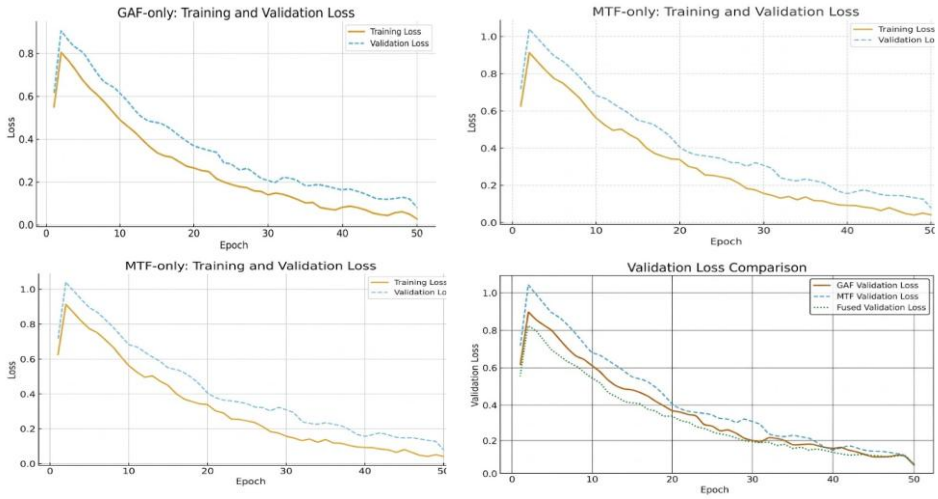

Figure 3. Training and validation loss for different model representations

5. CONCLUSION

This paper has proposed a CNN-based framework for the detection of CTC using a fusion-based GAF and MTF representation of IAT sequences. The results have shown the efficacy of the proposed technique in achieving high accuracy, above 96%, and outperforming the use of a single representation. Statistical tests have shown the significance of the improvements, ruling out the possibility of the results being due to random variations. Error metrics have also shown the stability and reliability of the proposed technique.

It is important to note that this work is specifically focused on offline classification in a controlled experimental scenario. Although the data was collected in a real network scenario, the covert encoding scheme is structured in a specific way. Therefore, the results are to be used as a baseline evaluation of this proposed approach. Future work will extend this study to experiment with the proposed approach under different network conditions, such as varying network traffic, noise, covert communication, and its applicability in different scenarios.

REFERENCES

[1] Al-Eidi, S., Darwish, O., Chen, Y. (2020). Covert timing channel analysis either as cyber attacks or confidential applications. Sensors, 20(8): 2417. https://doi.org/10.3390/s20082417

[2] Han, J., Huang, C., Shi, F., Liu, J. (2020). Covert timing channel detection method based on time interval and payload length analysis. Computers & Security, 97: 101952. https://doi.org/10.1016/j.cose.2020.101952

[3] Li, H., Song, T., Yang, Y. (2022). Generic and sensitive anomaly detection of network covert timing channels. IEEE Transactions on Dependable and Secure Computing, 20(5): 4085-4100. https://doi.org/10.1109/TDSC.2022.3207573

[4] Yazykova, A., Finoshin, M., Kogos, K. (2020). Artificial intelligence to detect timing covert channels. In BICA 2019: Proceedings of the Tenth Annual Meeting of the BICA Society, pp. 608-614. https://doi.org/10.1007/978-3-030-25719-4_79

[5] Elsadig, M.A., Gafar, A. (2022). Covert channel detection: Machine learning approaches. IEEE Access, 10: 38391-38405. https://doi.org/10.1109/ACCESS.2022.3164392

[6] Al-Eidi, S., Darwish, O., Chen, Y., Husari, G. (2020). SnapCatch: Automatic detection of covert timing channels using image processing and machine learning. IEEE Access, 9: 177-191. https://doi.org/10.1109/ACCESS.2020.3046234

[7] Al-Eidi, S., Darwish, O., Chen, Y., Elkhodr, M. (2022). Covert timing channels detection based on image processing using deep learning. In Proceedings of the 36th International Conference on Advanced Information Networking and Applications (AINA-2022), pp. 546-555. https://doi.org/10.1007/978-3-030-99619-2_51

[8] Al-Eidi, S., Darwish, O., Chen, Y., Maabreh, M., Tashtoush, Y. (2024). A deep learning approach for detecting covert timing channel attacks using sequential data. Cluster Computing, 27(2): 1655-1665. https://doi.org/10.1007/s10586-023-04035-5

[9] Sun, C., Chen, Y., Tian, H., Wu, S. (2021). Covert timing channels detection based on Auxiliary Classifier Generative Adversarial Network. IEEE Open Journal of the Computer Society, 2: 407-418. https://doi.org/10.1109/OJCS.2021.3131598

[10] Darwish, O., Al-Eidi, S., Al-Shorman, A., Maabreh, M., Alsobeh, A., Zahariev, P., Tashtoush, Y. (2026). LinguTimeX: A framework for multilingual CTC detection using explainable AI and natural language processing. Computers, Materials & Continua, 86(1): 1-21. https://doi.org/10.32604/cmc.2025.068266

[11] Javaid, A., Niyaz, Q., Sun, W., Alam, M. (2016). A deep learning approach for network intrusion detection system. EAI Endorsed Transactions on Security and Safety, 3(9): 21. https://doi.org/10.4108/eai.3-12-2015.2262516

[12] Wang, W., Zhu, M., Wang, J., Zeng, X., Yang, Z. (2017). End-to-end encrypted traffic classification with one-dimensional convolution neural networks. In 2017 IEEE International Conference on Intelligence and Security Informatics (ISI), Beijing, China, pp. 43-48. https://doi.org/10.1109/ISI.2017.8004872

[13] Meidan, Y., Bohadana, M., Shabtai, A., Ochoa, M., Tippenhauer, N.O., Guarnizo, J.D., Elovici, Y. (2017). Detection of unauthorized IoT devices using machine learning techniques. arXiv preprint arXiv:1709.04647. https://doi.org/10.48550/arXiv.1709.04647

[14] Moustafa, N., Slay, J. (2015). UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In 2015 Military Communications and Information Systems Conference (MilCIS), Canberra, ACT, Australia, pp. 1-6. https://doi.org/10.1109/MilCIS.2015.7348942

[15] Sharafaldin, I., Lashkari, A.H., Ghorbani, A.A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. In Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP 2018), pp. 108-116. https://doi.org/10.5220/0006639801080116

[16] Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y. (2014). Generative adversarial nets. Communications of the ACM, 3(11): 27. https://doi.org/10.1145/3422622

[17] Lin, Z., Shi, Y., Xue, Z. (2022). IDSGAN: Generative adversarial networks for attack generation against intrusion detection. In Advances in Knowledge Discovery and Data Mining. https://doi.org/10.1007/978-3-031-05981-0_7

[18] Rigaki, M., Garcia, S. (2018). Bringing a GAN to a knife-fight: Adapting malware communication to avoid detection. In 2018 IEEE Security and Privacy Workshops (SPW), San Francisco, CA, USA, pp. 70-75. https://doi.org/10.1109/SPW.2018.00019

[19] Mari, A.G., Zinca, D., Dobrota, V. (2023). Development of a machine-learning intrusion detection system and testing of its performance using a generative adversarial network. Sensors, 23(3): 1315. https://doi.org/10.3390/s23031315

[20] Wang, Z., Oates, T. (2015). Encoding time series as images for visual inspection and classification using tiled convolutional neural networks. In Workshops at the Twenty-Ninth AAAI Conference on Artificial Intelligence.

[21] Wang, Z., Oates, T. (2015). Imaging time series to improve classification and imputation. arXiv preprint arXiv:1506.00327.

https://doi.org/10.48550/arXiv.1506.00327
[22] Vargas, V.M., Ayllón-Gavilán, R., Durán-Rosal, A.M., Gutiérrez, P.A., Hervás-Martínez, C., Guijo-Rubio, D. (2023). Gramian angular and Markov Transition Fields applied to time series ordinal classification. In Advances in Computational Intelligence: 17th International Work-Conference on Artificial Neural Networks, IWANN 2023, Ponta Delgada, Portugal, pp. 505-516. https://doi.org/10.1007/978-3-031-43078-7_41
[23] Costa, H.V., Ribeiro, A.G., Souza, V.M. (2024). Fusion of image representations for time series classification with deep learning. In Artificial Neural Networks and Machine Learning – ICANN 2024: 33rd International Conference on Artificial Neural Networks, Lugano, Switzerland, pp. 235-250. https://doi.org/10.1007/978-3-031-72347-6_16
[24] Yang, C.L., Yang, C.Y., Chen, Z.X., Lo, N.W. (2019). Multivariate time series data transformation for convolutional neural networks. In 2019 IEEE/SICE International Symposium on System Integration (SII), pp. 188-192. https://doi.org/10.1109/SII.2019.8700425
[25] Wang, K., Li, K., Zhou, L., Hu, Y., Cheng, Z., Liu, J., Chen, C. (2019). Multiple convolutional neural networks for multivariate time series prediction. Neurocomputing, 360: 107-119. https://doi.org/10.1016/j.neucom.2019.05.023