Electronics Letters

LETTER OPEN ACCESS

Mixer-Based Neural Network for Binary Sequence Generation

Bora Yoon[1] | Junghyun Kim[1,2] | Hyojeong Choi[3] | Hong-Yeop Song[3]

[1]Department of Artificial Intelligence and Data Science, Sejong University, Seoul, Republic of Korea | [2]Deep Learning Architecture Research Center, Sejong University, Seoul, Republic of Korea | [3]Department of Electrical and Electronic Engineering, Yonsei University, Seoul, Republic of Korea

Correspondence: Junghyun Kim (j.kim@sejong.ac.kr)

Received: 15 May 2026 | **Revised:** 21 July 2026 | **Accepted:** 2 August 2026

ABSTRACT

This paper proposes MixBinNet, a neural network-based model that generates binary sequences with low autocorrelation sidelobes and low cross-correlation while maintaining a balanced bit distribution. Conventional sequence design methods lack flexibility in sequence length and set size, while often failing to jointly optimize correlation and balance properties. To overcome these limitations, MixBinNet employs an embedding block for diverse initialization, intra- and inter-mixing blocks that learn local and global correlation patterns and a binarization block for converting real-valued outputs into binary sequences. In addition, we design custom loss functions that enable the joint optimization of autocorrelation sidelobes, cross-correlation and balanced bit distributions during training. Experimental results show that MixBinNet consistently outperforms existing chaotic map-based methods in terms of autocorrelation and cross-correlation, while maintaining a bit distribution close to the ideal ratio of 0.5. These results demonstrate the effectiveness of MixBinNet in generating high-quality binary sequence sets.

1 | Introduction

Binary sequences are widely used in various signal processing systems including communications, radar and sensing for functions such as interference suppression and signal synchronization [1]. In multi-user and multipath environments, low autocorrelation and cross-correlation are essential for reducing interference and improving signal discrimination. Accordingly, code division multiple access (CDMA) systems employ sequences with high mutual orthogonality, such as pseudo-noise (PN) sequences [2], Gold codes [3] and Kasami codes [4]. Radar and positioning systems similarly require favorable autocorrelation properties to improve range resolution and target detection accuracy. Recently, various sequence families, including Barker codes [5], Frank codes [6] and Zadoff–Chu (ZC) sequences [7], have been widely investigated, together with mathematical optimization and search-based approaches for correlation reduction.

However, simultaneously achieving low autocorrelation sidelobes, low cross-correlation and balanced bit distributions remains challenging due to the inherent trade-offs among these properties and the rapidly increasing combinatorial search space for longer sequences or larger sets [8]. Consequently, practical systems often adopt simpler sequences at the cost of degraded performance [9]. Related limitations arise in direct sequence spread spectrum (DSSS) systems, where the finite number and periodic structure of conventional PN sequences can restrict code availability. In security-oriented applications, additional requirements such as aperiodicity and unpredictability may also be desirable.

To address these demands, chaotic map-based approaches have been widely explored because their sensitivity to initial conditions enables the generation of diverse sequences with favourable correlation properties [10–13]. Recent studies have further expanded the applications of chaotic maps to image encryption, information protection, hyperchaotic system design and memristive nonlinear systems, demonstrating their versatility in modern cryptographic and secure communication applications [14–17]. Despite these advances, most existing chaotic map-based approaches still

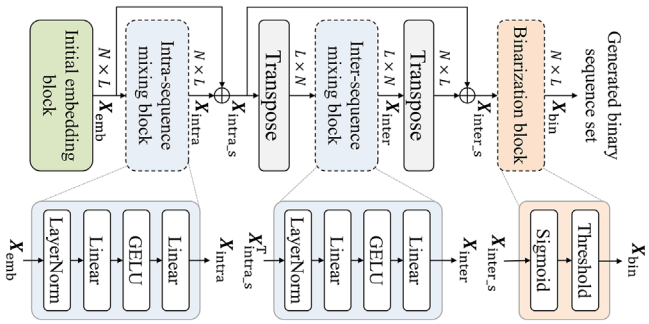


FIGURE 1 | Architecture of the proposed MixBinNet.

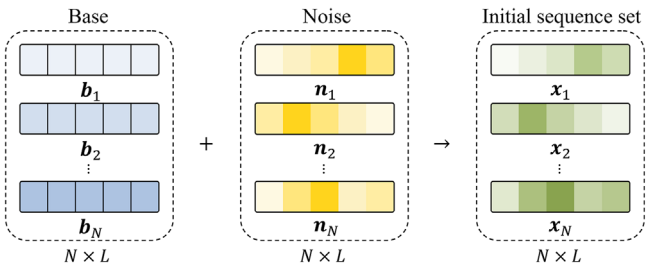


FIGURE 2 | Architecture of the initial embedding block of MixBinNet.

rely on predefined mathematical models, limiting their ability to directly optimize correlation properties for specific design objectives. Recently, neural network-based generative models have also attracted attention for their ability to learn data structures and directly optimize task-specific objectives [18, 19]. These characteristics make deep learning a promising framework for binary sequence design with explicit correlation and balance constraints.

Motivated by these advances, we propose MixBinNet, a neural network-based framework designed to generate binary sequence sets that satisfy predefined design objectives, including low autocorrelation, low cross-correlation and balanced bit distribution. MixBinNet consists of an embedding block for initial sequence diversity, intra- and inter-sequence mixing blocks for learning correlation structures and a binarization block for converting real-valued outputs into binary sequences. By jointly modeling intra- and inter-sequence correlation information, the proposed architecture generates binary sequences with low autocorrelation sidelobes and cross-correlation, while maintaining balanced bit distributions. Furthermore, we design custom loss functions that jointly suppress autocorrelation and cross-correlation while explicitly promoting bit balance, leading to stable training and improved generalization. Experimental results demonstrate that MixBinNet consistently outperforms chaotic map-based methods and a representative search-based optimization baseline, achieving lower correlation values and producing bit distributions closer to the ideal balance of 0.5.

2 | Proposed MixBinNet

The proposed MixBinNet is a neural network-based binary sequence generation model designed to facilitate the learning of binary sequences with low autocorrelation sidelobes and low cross-correlation, as illustrated in Figure 1. Bit balance is encouraged through dedicated training objectives, rather than being explicitly enforced by the network architecture. To promote sequence diversity, the architecture incorporates an embedding block that generates learnable initial sequences. Since the proposed framework performs gradient-based optimization in the real-valued domain prior to straight-through estimator (STE)-based binarization, a real-valued initialization is adopted instead of a standard random binary initialization. Specifically, the learnable sequence embeddings are initialized using zero-centered Gaussian logits. This initialization keeps the initial sigmoid outputs close to the decision boundary, preventing premature saturation of the sigmoid activation while promoting balanced binary probabilities. In addition, Gaussian perturbations introduce sufficient diversity into the initial sequence representations. Accordingly, the initial embedding matrix is obtained by adding Gaussian noise $\boldsymbol{n}_i$ to a zero-valued base matrix in the real-valued domain. The resulting embedding matrix serves as the learnable initialization of the proposed framework and can be flexibly generated for arbitrary numbers of sequences N and sequence lengths L. This process can be expressed as:

$$\boldsymbol{b}_i = \text{Zeros}(L), \tag{1}$$

$$\boldsymbol{x}_i = \boldsymbol{b}_i + \boldsymbol{n}_i, \tag{2}$$

$$\boldsymbol{X}_{\text{emb}} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N], \tag{3}$$

where N is the number of sequences, and $\boldsymbol{x}_i$ denotes the generated initial sequence. As illustrated in Figure 2, this procedure generates the initial sequence set $\boldsymbol{X}_{\text{emb}}$, and both N and L can be flexibly specified. Although the initial embeddings are treated as learnable parameters, they are not introduced to memorize predefined binary sequences. Instead, they are jointly optimized with the network parameters under the proposed objective functions to generate a binary sequence set satisfying the predefined design objectives.

While the embedding block provides a stable and balanced initialization, it does not explicitly model element-wise dependencies within each sequence. To address this limitation, the intra-sequence mixing block learns within-sequence interactions, resulting in lower autocorrelation. This block adopts a multilayer perceptron (MLP)-based architecture that applies nonlinear transformations along the sequence length L. Layer normalization (LN) is first applied to the input sequence $\boldsymbol{X}_{\text{emb}}$ to improve training stability, followed by a linear-Gaussian error linear unit (GELU)-linear sub-block that captures local correlation patterns. Within this sub-block, the initial linear layer expands the dimension from L to a hidden dimension d_{hid}, and the GELU activation introduces nonlinearity. The subsequent linear layer then projects the learned representations back to L. This process can be expressed as:

$$\boldsymbol{X}_{\text{intra}} = \boldsymbol{W}_2 \cdot \text{GELU}(\boldsymbol{W}_1 \cdot \text{LN}(\boldsymbol{X}_{\text{emb}})), \tag{4}$$

where $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$ denote learnable weight matrices. The output is then combined with the input sequence via a skip connection to prevent information loss and improve training stability, that is, $\boldsymbol{X}_{\text{intra\_s}} = \boldsymbol{X}_{\text{emb}} + \boldsymbol{X}_{\text{intra}}$.

Even with richer intra-sequence variation, different sequences may still converge to similar representations. To explicitly capture cross-sequence interactions and further reduce cross-correlation, we apply an inter-sequence mixing block. This block operates along the sequence set size N, applying nonlinear transformations to improve inter-sequence discriminability. Specifically, the output $\boldsymbol{X}_{\text{intra\_s}}$ is transposed, normalized across sequences, and then processed through a linear-GELU-Linear structure to model inter-sequence interactions, producing an intermediate representation $\boldsymbol{X}_{\text{inter}}$, defined as follows:

$$\boldsymbol{X}_{\text{inter}} = \boldsymbol{W}_4 \cdot \text{GELU}(\boldsymbol{W}_3 \cdot \text{LN}(\boldsymbol{X}_{\text{intra\_s}}^{\top})), \tag{5}$$

where $\boldsymbol{W}_3$ and $\boldsymbol{W}_4$ denote learnable weight matrices. The output $\boldsymbol{X}_{\text{inter}}$ is then transposed back to its original dimensionality, and a skip connection is applied to preserve information and improve training stability, that is, $\boldsymbol{X}_{\text{inter\_s}} = \boldsymbol{X}_{\text{intra\_s}} + \boldsymbol{X}_{\text{inter}}^{\top}$.

To convert the refined real-valued output into binary sequences, a binarization block is applied as the final stage. Specifically, a sigmoid function maps each real-valued output to the continuous domain $(0, 1)$, as follows:

$$\boldsymbol{X}_{\text{sig}} = \text{Sigmoid}(\boldsymbol{X}_{\text{inter\_s}}). \tag{6}$$

Afterward, a thresholding function applies hard binarization by mapping values above 0.5 to 1 and the rest to 0, as follows:

$$\boldsymbol{X}_{\text{bin}} = \text{Threshold}(\boldsymbol{X}_{\text{sig}}), \tag{7}$$

$$\text{Threshold}((\boldsymbol{X}_{\text{sig}})_{i,j}) = \begin{cases} 1, & \text{if } (\boldsymbol{X}_{\text{sig}})_{i,j} > 0.5, \\ 0, & \text{otherwise}, \end{cases} \tag{8}$$

where $i \in [1, N]$, $j \in [1, L]$. To enable gradient propagation through the threshold operation during training, a STE is employed, since hard binarization is non-differentiable. As the STE is applied only at the final binarization stage, the remaining network is optimized in the continuous domain. In our experiments, this design resulted in stable convergence without noticeable divergence or oscillation during training.

3 | Custom Loss Function Design

To promote desirable properties in the generated binary sequences, we define a set of loss functions tailored to the training process. In particular, correlation-based loss terms are introduced to regulate autocorrelation and cross-correlation characteristics, while additional terms are incorporated to encourage balanced bit distributions and sequence uniqueness. To facilitate correlation computation, each sequence in the final binary set $\boldsymbol{X}_{\text{bin}}$ is converted into a bipolar representation $\boldsymbol{v}_i$ by mapping binary values 0 and 1 to -1 and 1, respectively, as defined below:

$$\boldsymbol{v}_i = 2(\boldsymbol{X}_{\text{bin}})_i - 1. \tag{9}$$

Using the converted sequences, autocorrelation and cross-correlation metrics are computed and used as core components of the loss function. Since direct time-domain correlation is computationally expensive, a fast Fourier transform (FFT)-based approach using the convolution theorem [20] is adopted to improve efficiency. Specifically, autocorrelation is computed by applying the inverse Fourier transform to the product of the Fourier transform of each real-valued sequence and its complex conjugate, as defined below:

$$r_i[k] = \frac{1}{L}\mathcal{F}^{-1}\Big(\mathcal{F}(\boldsymbol{v}_i) \cdot \overline{\mathcal{F}(\boldsymbol{v}_i)}\Big)[k], \quad i = 1, 2, \ldots, N, \tag{10}$$

where $\overline{(\cdot)}$ denotes the complex conjugate operation, $\mathcal{F}^{-1}(\cdot)$ represents the inverse Fourier transform and $r_i[k]$ denotes the FFT-based autocorrelation of the ith sequence, where k represents the correlation shift.

To accurately evaluate the similarity among the generated sequences, the cross-correlation is computed over all possible sequence pairs. Prior to correlation computation, the mean of each sequence is removed to avoid correlation bias caused by a non-zero mean (i.e., a direct current (DC) component), which would otherwise inflate the correlation values. The mean-removed sequences and the corresponding cross-correlation are defined as follows:

$$\tilde{\boldsymbol{v}}_i = \boldsymbol{v}_i - \frac{1}{L}\sum_{k=0}^{L-1} \boldsymbol{v}_i[k], \tag{11}$$

$$c_{i,j}[k] = \frac{1}{L}\mathcal{F}^{-1}\Big(\mathcal{F}(\tilde{\boldsymbol{v}}_i) \cdot \overline{\mathcal{F}(\tilde{\boldsymbol{v}}_j)}\Big)[k], \tag{12}$$

where $\tilde{\boldsymbol{v}}_i$ denotes the mean-removed version of the ith sequence, and $c_{i,j}[k]$ represents the FFT-based cross-correlation between the ith and jth sequences.

Using the correlation metrics defined above as components of the loss function, the overall training objective is constructed to jointly optimize correlation and balance properties. The total loss function is defined as follows:

$$\begin{aligned} \mathcal{L} = {} & \alpha_1\mathcal{L}_{\text{AC,mean}} + \alpha_2\mathcal{L}_{\text{AC,peak}} + \alpha_3\mathcal{L}_{\text{CC,mean}} \\ & + \alpha_4\mathcal{L}_{\text{CC,peak}} + \alpha_5\mathcal{L}_{\text{CC,smooth}} + \alpha_6\mathcal{L}_{\text{CC,top-K}} \\ & + \alpha_7\mathcal{L}_{\text{CC,margin}} + \alpha_8\mathcal{L}_{\text{bal}} + \alpha_9\mathcal{L}_{\text{uniq}} + \mathcal{L}_{\text{reg}}, \end{aligned} \tag{13}$$

where $\{\alpha_1, \ldots, \alpha_9\}$ denote the weighting coefficients of the corresponding loss terms. During training, a curriculum-based weighting strategy is adopted by gradually increasing the weights of the cross-correlation-related loss terms, while keeping the autocorrelation loss weights fixed. This strategy progressively emphasizes highly correlated sequence pairs under the all-pair formulation.

First, two autocorrelation-based loss terms, $\mathcal{L}_{\text{AC,mean}}$ and $\mathcal{L}_{\text{AC,peak}}$, are introduced to mitigate periodic and repetitive structures in the generated sequences. Since the autocorrelation peak at $k = 0$ provides limited information for characterizing sequence properties, only sidelobe components are considered in the loss computation. Specifically, minimizing $\mathcal{L}_{\text{AC,mean}}$ and $\mathcal{L}_{\text{AC,peak}}$ jointly suppresses both the average level and locally dominant peaks of autocorrelation sidelobes during training.

$$\mathcal{L}_{\text{AC,mean}} = \frac{1}{N(L-1)}\sum_{i=1}^{N}\sum_{k=1}^{L-1} |r_i[k]|, \tag{14}$$

$$\mathcal{L}_{\mathrm{AC,peak}} = \frac{1}{N}\sum_{i=1}^{N}\max_{1\le k\le L-1}|r_i[k]|, \tag{15}$$

where $r_i[k]$ denotes the autocorrelation value of the ith sequence.

To address inter-sequence similarity, the proposed framework employs five complementary cross-correlation loss terms. The mean and maximum cross-correlation losses optimize the overall cross-correlation characteristics of the sequence set by reducing both the average correlation level and the dominant peak within each sequence pair. The smooth maximum, hard top-K and margin-based loss terms further focus the optimization on sequence pairs exhibiting large cross-correlation peaks. This complementary design enables more effective optimization under the all-pair cross-correlation formulation by suppressing both the overall cross-correlation level and locally dominant correlation peaks.

The mean cross-correlation loss minimizes the average cross-correlation over all sequence pairs and is defined as:

$$\mathcal{L}_{\mathrm{CC,mean}} = \frac{2}{N(N-1)L}\sum_{i=1}^{N-1}\sum_{j=i+1}^{N}\sum_{k=0}^{L-1}\left|c_{i,j}[k]\right|, \tag{16}$$

where $c_{i,j}[k]$ denotes the cross-correlation value between the ith and jth sequences.

To additionally suppress dominant peaks within each sequence pair, the maximum cross-correlation loss is defined as:

$$p_{i,j} = \max_{0\le k\le L-1}\left|c_{i,j}[k]\right|, \quad i<j, \tag{17}$$

$$\mathcal{L}_{\mathrm{CC,peak}} = \frac{2}{N(N-1)}\sum_{i=1}^{N-1}\sum_{j=i+1}^{N}p_{i,j}. \tag{18}$$

To further emphasize sequence pairs exhibiting large cross-correlation peaks while avoiding abrupt weighting changes, a smooth weighting strategy based on the softmax function is additionally employed.

$$\mathcal{L}_{\mathrm{CC,smooth}} = \sum_{i=1}^{N-1}\sum_{j=i+1}^{N}\frac{\exp\left(p_{i,j}/\tau_{\mathrm{s}}\right)}{\sum_{a=1}^{N-1}\sum_{b=a+1}^{N}\exp\left(p_{a,b}/\tau_{\mathrm{s}}\right)}p_{i,j}, \tag{19}$$

where τ_{s} denotes the softmax temperature controlling the degree of emphasis on larger peak cross-correlation values.

Although the previous loss terms reduce the overall cross-correlation and dominant peaks, they treat all sequence pairs equally. To place greater optimization emphasis on sequence pairs with the largest cross-correlation peaks, a hard top-K loss is introduced.

$$K = \max\left(1,\ \left\lfloor\frac{N(N-1)}{2}\rho\right\rfloor\right), \tag{20}$$

$$\mathcal{T}_K = \mathrm{TopK}\left(\{p_{i,j}\}_{i<j}, K\right), \tag{21}$$

$$\mathcal{L}_{\mathrm{CC,top\text{-}K}} = \frac{1}{K}\sum_{t\in\mathcal{T}_K}t, \tag{22}$$

where ρ denotes the hard top-K ratio, K is the number of selected peak cross-correlation values determined by ρ and $\mathcal{T}_K$ denotes the set of the K largest peak cross-correlation values selected from $\{p_{i,j}\}_{i<j}$.

Finally, a margin-based loss penalizes only sequence pairs whose peak cross-correlation exceeds a predefined margin, thereby avoiding unnecessary optimization of sequence pairs that already satisfy the desired correlation constraint.

$$\mathcal{L}_{\mathrm{CC,margin}} = \frac{2}{N(N-1)}\sum_{i=1}^{N-1}\sum_{j=i+1}^{N}\left[\max\left(0, p_{i,j}-\tau_{\mathrm{m}}\right)\right]^2, \tag{23}$$

where τ_{m} denotes the predefined margin threshold.

While correlation-based loss terms effectively suppress inter-sequence interference and repetitive structures, they do not directly enforce bit balance. Therefore, a balance loss term $\mathcal{L}_{\mathrm{bal}}$ is incorporated to explicitly regulate the bit distribution. A straightforward approach is to regulate the global proportion of ones across the entire sequence set toward 0.5. However, satisfying this global criterion alone does not guarantee a balanced bit distribution within each individual sequence. Conversely, optimizing only the row-wise balance may still result in a biased global bit distribution across the sequence set. Therefore, the proposed framework jointly employs global and row-wise balance loss terms. Since row-wise balance directly regulates the bit distribution within each individual sequence and more effectively mitigates sequence-level bit imbalance, it is assigned a larger weighting factor than the global balance term. The balance loss is defined as:

$$\mathcal{L}_{\mathrm{bal}} = \lambda_{\mathrm{g}}\left(0.5-\frac{1}{NL}\sum_{i=1}^{N}\sum_{k=0}^{L-1}s_i[k]\right)^2 + \lambda_{\mathrm{r}}\frac{1}{N}\sum_{i=1}^{N}\left|0.5-\frac{1}{L}\sum_{k=0}^{L-1}s_i[k]\right|, \tag{24}$$

where $s_i[k]$ denotes the kth bit of the ith sequence. The first term penalizes deviations of the global bit ratio from the desired value of 0.5, while the second term encourages each individual sequence to maintain a balanced proportion of ones. Here, λ_{g} and λ_{r} denote the weighting factors for the global and row-wise balance loss terms, respectively. In this work, $\lambda_{\mathrm{g}} = 0.25$ and $\lambda_{\mathrm{r}} = 0.75$.

Additionally, a uniqueness loss term $\mathcal{L}_{\mathrm{uniq}}$ is introduced as an auxiliary constraint to discourage multiple learnable sequence representations from converging to identical or highly similar solutions during joint optimization, thereby mitigating optimization-induced sequence collapse and within the generated sequence set. This term is defined based on the complement of the ratio of unique sequences to the total number of sequences, as follows:

$$\mathcal{L}_{\mathrm{uniq}} = 1-\frac{N_{\mathrm{uniq}}}{N}. \tag{25}$$

Finally, to mitigate overfitting and improve training stability, an L2 regularization term $\mathcal{L}_{\mathrm{reg}}$ is applied to the embedding parameters $\boldsymbol{W}_{\mathrm{emb}}$, as follows:

$$\mathcal{L}_{\mathrm{reg}} = \lambda_{\mathrm{reg}}\left\|\boldsymbol{W}_{\mathrm{emb}}\right\|_2, \tag{26}$$

TABLE 1 | Comparison of correlation properties and bit balance among binary sequence generation methods ($N = 1000$, $L = 2048$).

	Autocorrelation		Cross-correlation		
Type	**Average sidelobe**	**Peak average sidelobe**	**Average**	**Peak average**	**Ratio of 1's**
Logistic map [10]	0.01761	0.07597	0.01764	0.08006	0.49913
Tent map [11]	0.01760	0.07567	0.01763	0.08008	0.50040
Chebyshev map [12]	0.01758	0.07622	0.01763	0.08007	0.50063
GA	0.01760	0.07523	0.01763	0.08005	0.49976
Proposed	**0.01739**	**0.07139**	**0.01763**	**0.07938**	**0.49981**

Note: Boldface indicates the best performance for each metric.

TABLE 2 | Comparison of correlation properties and bit balance among binary sequence generation methods ($N = 500$, $L = 1024$).

	Autocorrelation		Cross-correlation		
Type	**Average sidelobe**	**Peak average sidelobe**	**Average**	**Peak average**	**Ratio of 1's**
Logistic map [10]	0.02490	0.10089	0.02493	0.10750	0.49880
Tent map [11]	0.02497	0.10170	0.02493	0.10744	0.49920
Chebyshev map [12]	0.02492	0.10159	0.02493	0.10746	0.50113
GA	0.02489	0.10068	0.02493	0.10726	0.50022
Proposed	**0.02450**	**0.09754**	**0.02493**	**0.10672**	**0.49999**

Note: Boldface indicates the best performance for each metric.

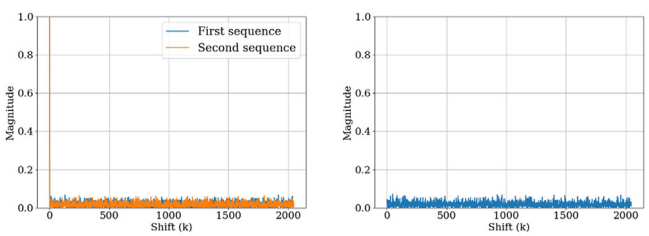
where λ_{reg} denotes the weighting coefficient of the regularization term, which is set to $\lambda_{\text{reg}} = 2 \times 10^{-5}$ in this paper.

All the loss terms introduced above are integrated into a unified optimization objective, enabling the proposed framework to jointly optimize correlation properties, bit balance and sequence diversity during training.

4 | Experimental Results

To evaluate the correlation and balance properties of the binary sequences generated by MixBinNet, we compared it with chaotic map-based sequence generation methods and a genetic algorithm (GA)-based optimization baseline. For fair comparison, the GA baseline was configured to optimize the same autocorrelation, cross-correlation and bit balance objectives as the proposed framework. Sequences generated in the chaotic regime with strong aperiodicity and randomness were used as baselines. All experiments were performed with a sequence length of $L = 2048$ and $N = 1000$ sequences. To evaluate the generality of the proposed framework, additional experiments were also conducted with $L = 1024$ and $N = 500$ using the same architecture and hyperparameter settings without further tuning. MixBinNet was trained for 300 epochs using the AdamW optimizer with an initial learning rate of 3×10^{-4}. A cosine annealing scheduler with a minimum learning rate of 5×10^{-5} was employed. The initial weighting coefficients were empirically set to $(\alpha_1, \alpha_2, \ldots, \alpha_9) = (2, 1, 2, 4, 2, 4, 4, 9, 5)$.

Tables 1 and 2 compare the correlation and bit balance performance of MixBinNet with chaotic map-based methods and a GA-based optimization baseline under two sequence configurations.



(a) Autocorrelation magnitudes of each sequence

(b) Cross-correlation magnitude between the two sequences

FIGURE 3 | Autocorrelation and cross-correlation of two representative binary sequences generated by MixBinNet: (a) autocorrelation magnitudes of each sequence and (b) cross-correlation magnitude between the two sequences.

The average metric denotes the mean magnitude of the autocorrelation sidelobes or cross-correlation values, whereas the peak average metric represents the mean of the corresponding peak autocorrelation sidelobes or peak cross-correlation values. For both configurations, MixBinNet achieves the lowest peak cross-correlation and the best autocorrelation performance among all compared methods. In addition, the bit ratio remains consistently close to the ideal value of 0.5, demonstrating that the generated sequences are well balanced. The consistent performance observed for both $N = 1000, L = 2048$ and $N = 500, L = 1024$, using the same architecture and training settings, suggests that the proposed framework can be applied to different sequence lengths and set sizes without additional hyperparameter tuning. Figure 3 illustrates the autocorrelation of the first and second sequences generated by MixBinNet, along with their cross-correlation, showing a sharp autocorrelation peak at zero lag and low cross-correlation levels.

These performance improvements require an offline training process. However, this one-time training cost enables the direct optimization of binary sequence sets for user-defined autocorrelation, cross-correlation and bit balance objectives, unlike conventional chaotic map-based methods that rely on predefined mathematical dynamics. The same framework can also be readily applied to different sequence lengths, sequence set sizes and design objectives without modifying the network architecture. Once trained, the optimized sequence set can be directly used without additional optimization.

The training experiments were conducted on an NVIDIA GeForce RTX 5080 GPU. Under this hardware configuration, the average training time was approximately 216 s per epoch, resulting in a total training time of approximately 64,800 s (about 18.0 h) for 300 epochs. The peak GPU memory consumption during training was approximately 8.5 GB.

5 | Conclusion

This paper presented MixBinNet, a neural network-based framework for generating binary sequence sets with well-controlled correlation characteristics and balanced bit distributions. By leveraging learnable representations and correlation-aware training objectives, the proposed approach enables the generation of sequences with low autocorrelation sidelobes and reduced cross-correlation while maintaining balanced bit distributions. Experimental evaluations demonstrated that MixBinNet consistently outperforms both chaotic map-based methods and the GA-based optimization baseline. Furthermore, the proposed framework maintained its performance across different sequence lengths and sequence set sizes using the same architecture and training settings. These results highlight the potential of MixBinNet as a flexible and effective solution for binary sequence generation, with applicability to radar systems, spread-spectrum communications and synchronization tasks.

Author Contributions

Bora Yoon: conceptualization, methodology, validation, writing – original draft, visualization. **Junghyun Kim:** conceptualization, methodology, validation, writing – review & editing, supervision. **Hyojeong Choi:** writing – review & editing. **Hong-Yeop Song:** writing – review & editing.

Acknowledgements

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) under the virtual convergence support program to nurture the best talents (IITP-2026-RS-2023-00254529) grant funded by the Korea government (MSIT). This work was also supported by the National Research Foundation of Korea (NRF) Grant by the Korean Government through the Ministry of Sciences and ICT (MSIT) under Grant RS-2023-00209000.

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

Data sharing is not applicable to this article as no datasets were generated or analysed during the current study.

References

1. O. Rezaei, M. Ahmadi, M. M. Naghsh, A. Aubry, M. M. Nayebi, and A. De Maio, "A Learning-Inspired Strategy to Design Binary Sequences with Good Correlation Properties: SISO and MIMO Radar Systems," *IEEE Transactions on Aerospace and Electronic Systems* 59, no. 5 (2023): 6637–6657.

2. S. W. Golomb and G. Gong, *Signal Design for Good Correlation: For Wireless Communication, Cryptography, and Radar* (Cambridge University Press, 2005).

3. R. Gold, "Maximal Recursive Sequences with 3-Valued Recursive Cross-Correlation Functions (Corresp.)," *IEEE Transactions on Information Theory* 14, no. 1 (1968): 154–156.

4. J. Lahtonen, "On the Odd and the Aperiodic Correlation Properties of the Kasami Sequences," *IEEE Transactions on Information Theory* 41, no. 5 (1995): 1506–1508.

5. J. Soba, A. Munir, and A. B. Suksmono, "Barker Code Radar Simulation for Target Range Detection Using Software Defined Radio," in *Proceedings of the International Conference on Information Technology and Electrical Engineering (ICITEE)* (IEEE, 2013), 271–276.

6. R. L. Frank, "Polyphase Codes with Good Nonperiodic Correlation Properties," *IRE Transactions on Information Theory* 9, no. 1 (1963): 43–45.

7. D. C. Chu, "Polyphase Codes with Good Periodic Correlation Properties," *IEEE Transactions on Information Theory* 18, no. 4 (1972): 531–532.

8. L. Jin, R. Zhu, and C. Xing, "Binary Sequences with a Low Correlation via Cyclotomic Function Fields," *IEEE Transactions on Information Theory* 68, no. 5 (2022): 3445–3454.

9. L. Jin, L. Ma, C. Xing, and R. Zhu, "A New Family of Binary Sequences with a Low Correlation via Elliptic Curves," *IEEE Transactions on Information Theory* 71, no. 8 (2025): 6470–6478.

10. H. Jiang and C. Fu, "A Chaos-Based High Quality PN Sequence Generator," in *Proceedings of the International Conference on Intelligent Computation Technology and Automation (ICICTA)* (IEEE, 2008), 60–64.

11. C. Li, G. Luo, K. Qin, and C. Li, "An Image Encryption Scheme Based on Chaotic Tent Map," *Nonlinear Dynamics* 87, no. 1 (2017): 127–133.

12. F. Liu, S. Jia, X. Xu, and M. Tian, "Improved Chaotic Sequence Generation Method Based on Direct Spread Spectrum," *Journal of Physics: Conference Series* 1237, no. 4 (2019): 042006.

13. H. Choi, G. Kim, H.-Y. Song, and H. Noh, "Chaotic Nature of Integer Sequences From Primitive Linear Feedback Shift Registers," *IEEE Transactions on Circuits and Systems II: Express Briefs* 72, no. 9 (2025): 1268–1272.

14. X. Shi, Y. Su, and X. Yang, "Lossless Frequency-Domain Image Encryption via 3D Exponential Hyper-Chaotic Map and Integer Lifting Wavelet Transform," *Axioms* 15, no. 5 (2026): 315, 10.3390/axioms15050315.

15. W. Feng, Z. Tang, X. Zhao, et al., "State-Dependent Variable Fractional-Order Hyperchaotic Dynamics in a Coupled Quadratic Map: A Novel System for High-Performance Image Protection," *Fractal and Fractional* 9, no. 12 (2025): 792, 10.3390/fractalfract9120792.

16. F. Yu, X. Wang, Y. Xiao, et al., "Extreme Multistability in Discrete Memristive Neuron Maps and Implications for Dual-Field Applications," *Integration* 109 (2026): 102688, 10.1016/j.vlsi.2026.102688.

17. W. Feng, Z. Tang, X. Zhao, et al., "Two-Dimensional Coupling-Enhanced Cubic Hyperchaotic Map with Exponential Parameters: Construction, Analysis, and Application in Hierarchical Significance-Aware Multi-Image Encryption," *Axioms* 14, no. 12 (2025): 901, 10.3390/axioms14120901.

18. I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, 2016).

19. H. Lee, B. Lee, H. Yang, et al., "Towards 6G Hyper-Connectivity: Vision, Challenges, and Key Enabling Technologies," *Journal of Communications and Networks* 25, no. 3 (2023): 344–354.

20. A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing* (Pearson Education India, 1999).