CS-Net: A Deep Learning-Based Analysis for the Cryptography and Steganography

Hansung University
Dukyoung Kim

한성대학교
HANSUNG UNIVERSITY

CryptoCraft LAB

Contents

Motivation

- **Limitations of Individual Methods.**
  - Steganography conceals data but is vulnerable if its method is exposed, while cryptography secures data without hiding its existence.

- **Need for Combined Security Analysis.**
  - Combining steganography and cryptography offers enhanced security, yet deep learning models to analyze this integration are underexplored.

- **Introducing CS-Net.**
  - To address this gap, we propose CS-Net, a model for analyzing data secured through both techniques, aiming to improve secure communication.

Contributions

- **Deep Learning-Based Analysis of Combined Cryptography and Steganography.**
  - First deep learning approach to analyze combined cryptography and steganography.
  - Enabling new possibilities in data security.

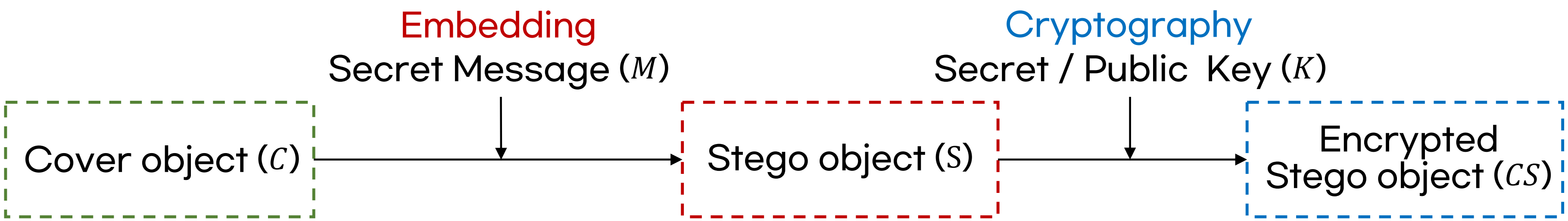
- **Development of CS-Net Model with High Accuracy.**
  - CS-Net reliably identifies encrypted stego images.
  - A robust framework for integrated security analysis.

- **Advanced Preprocessing and Rotation-Based Learning Technique.**
  - It is effective technique even with cryptography.

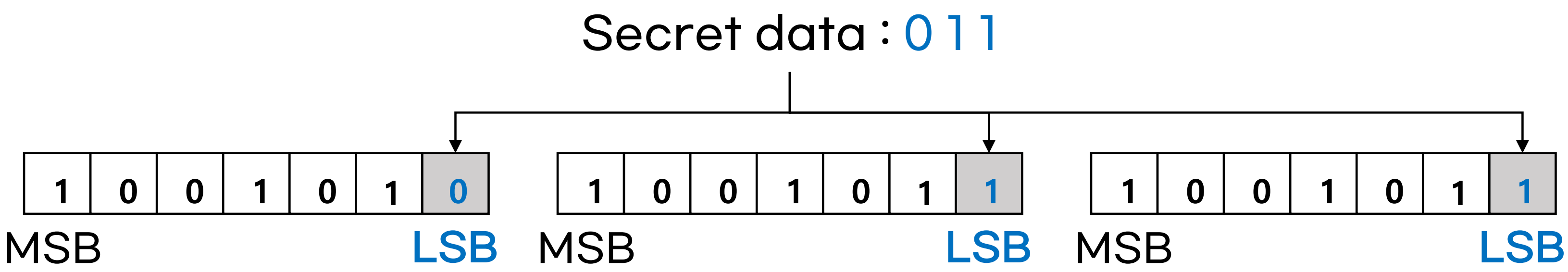
Background

Steganography

- **Steganography**
  - A technique for hiding secret messages within digital media.
  - It makes detection of the secret data hard.

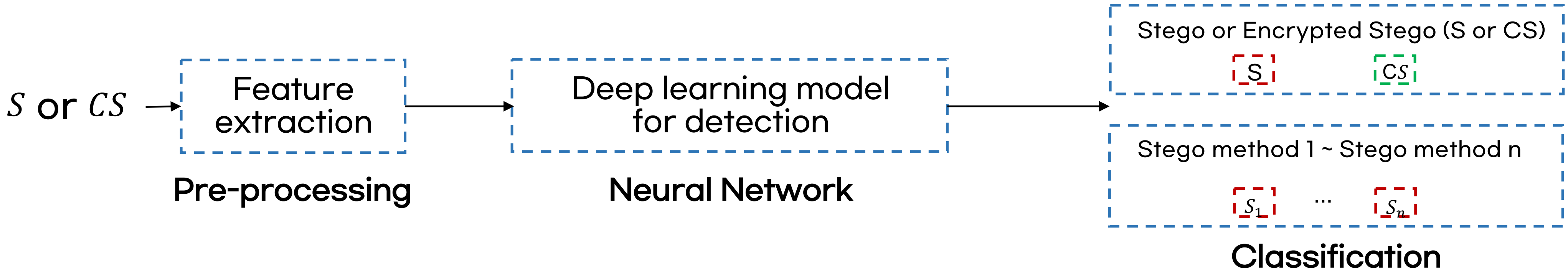


- **LSB Steganography**
  - It hides data by altering the least significant bits (LSB) of an image.
  - CS-Net applies LSB method with encryption for enhanced security.



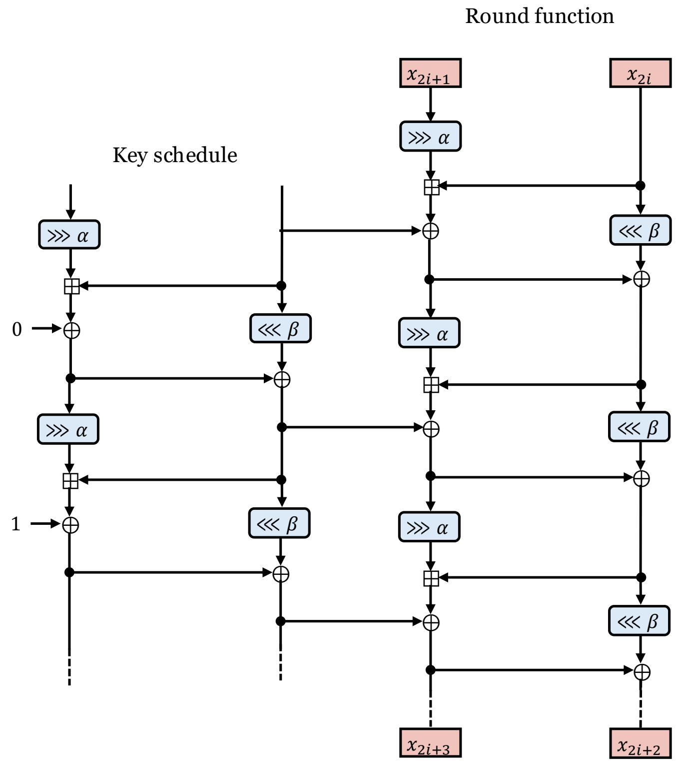
Steganalysis

- **Steganalysis**
  - Detecting hidden data in digital media by analyzing patterns and signals.
  - If the steganography technique is known, it can be uncovered by reversing the logic.
- Deep learning-based Steganalysis (Classification)
  - A deep learning model can **detect the steganography method used.**
  - Preprocessing: Extracting features from stego data (pure or encrypted image)
  - Detection: Classifying the embedding method using preprocessed data as input



SPECK

- **SPECK**
  - A lightweight symmetric key cipher developed by the NSA.
- **SPECK has multiple variants**
  - Block sizes: **32**, 48, 64, 96, and 128 bits
  - Key sizes: **64**, 72, 96, 128, 144, 192, and 256 bits
- **SPECK Encryption Process**
  - Uses rotation, addition, and XOR to mix data effectively.



Schematic of SPECK encryption

Related Work Comparison

- **Previous Research**
  - Previous studies focused on enhancing steganography by increasing embedding com plexity, without integrating encryption.

- **CS-Net's Distinction**
  - CS-Net is the first deep learning model to analyze encrypted stego data using SPECK encryption, offering a new direction in secure data analysis.

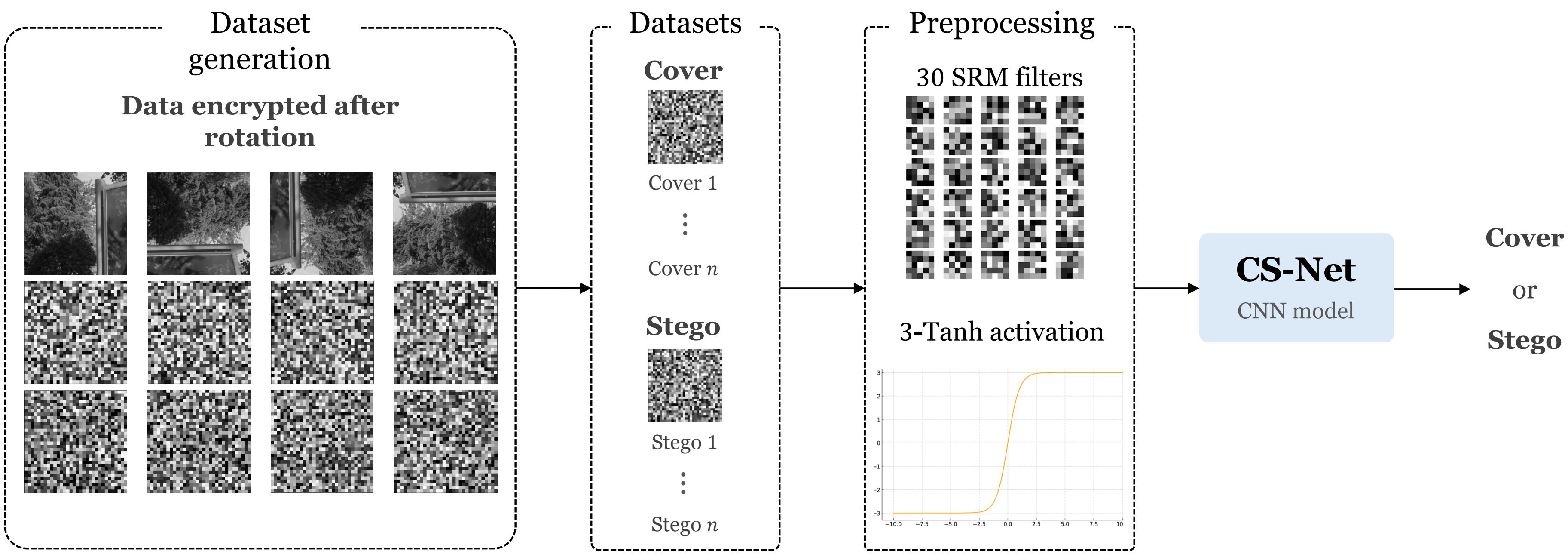
Framework	Steganography	Cryptography
Xu-Net [6] Ye-Net [7] Yedroudj-Net [8] GBRAS-Net [9]	✓ (WOW, S-UNIWARD)	✗
CS-Net (Ours)	✓ (LSB)	✓ (SPECK)

Related Work Comparison.

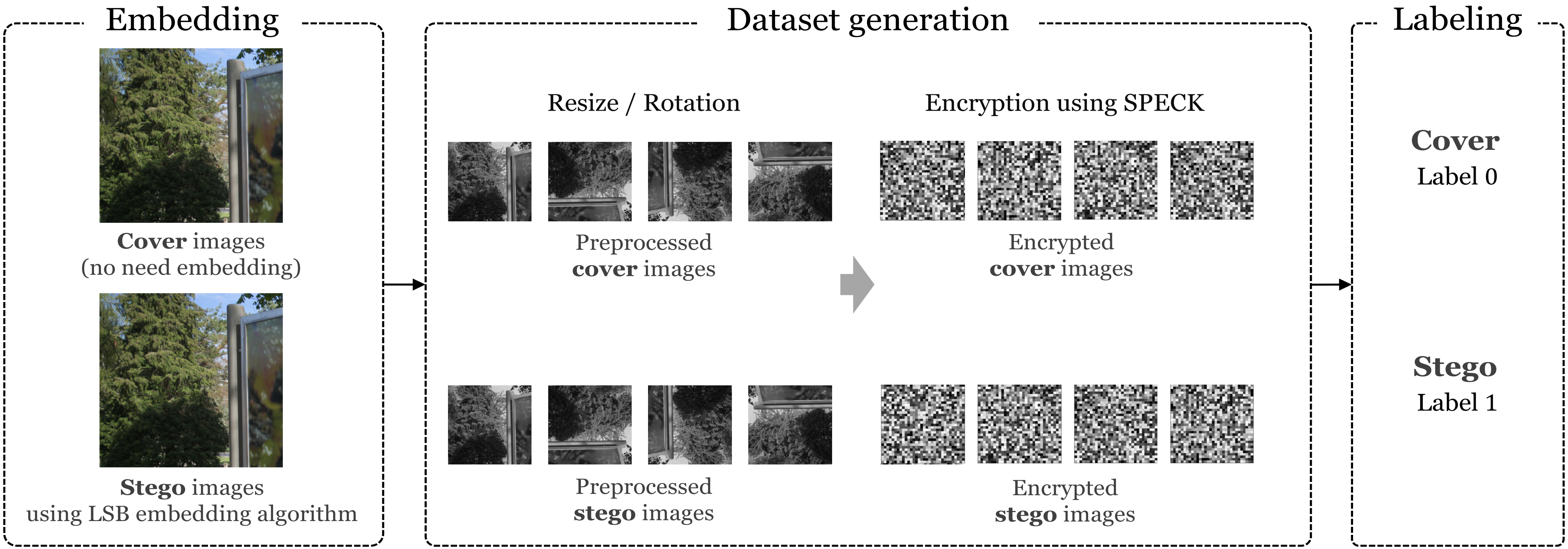
Proposed Method

Overview of CS-Net

- A deep learning model that classifies encrypted images as **cover or stego.**
- It combines **LSB steganography, SPECK encryption, and a rotation strategy** to improve learning.
- To accurately detect hidden data in encrypted images.



Dataset: Generation Process



LSB

We choose basic algorithm
to apply encryption.

Rotation

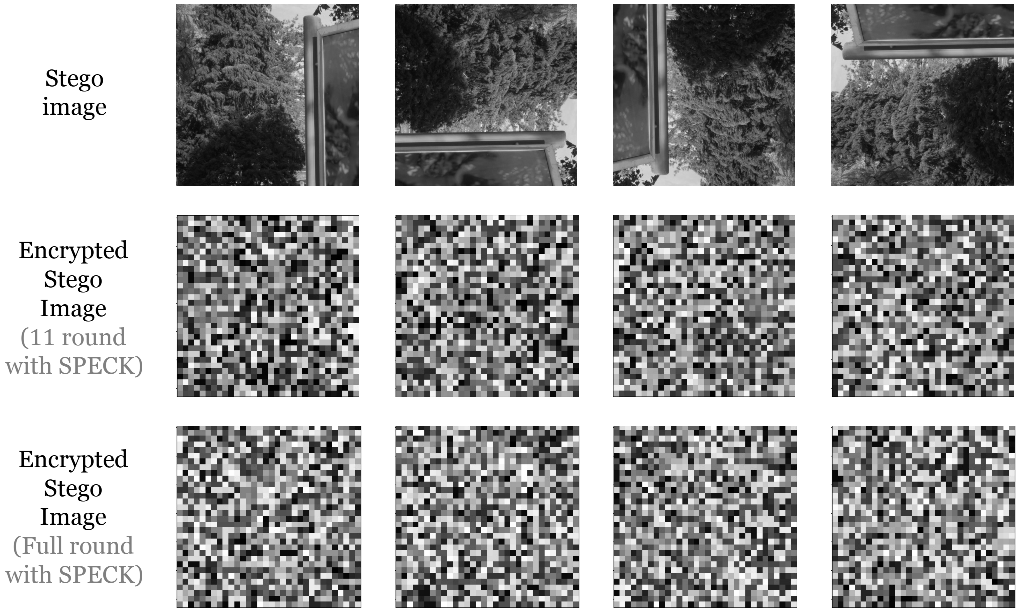
For considering
different data perspectives

Apply SPECK for confidentiality

Cover (with SPECK only)

Stego (with LSB+SPECK)

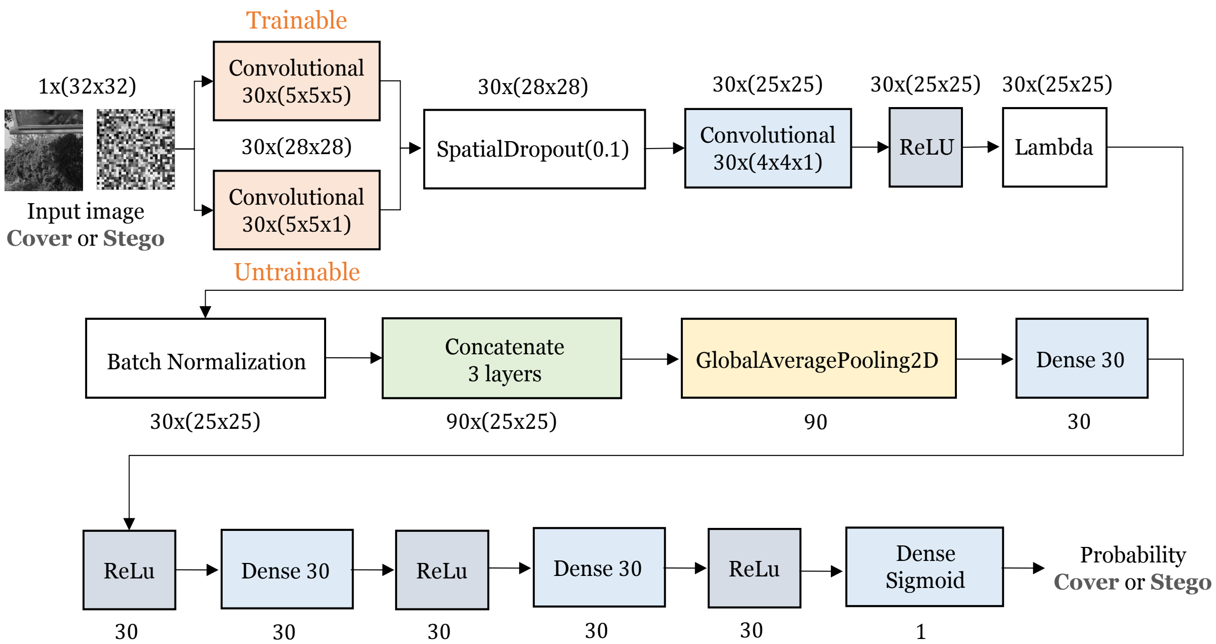
Dataset: Novel Rotation Strategy



Stego and encrypted 32 × 32 images using the rotation strategy.

- **Data Augmentation**
  - Rotate stego images by 90°, 180°, and 270° for diverse training data.
- **Multi-Directional Encryption**
  - Encrypts 32-bit blocks in various directions (left, right, down, up).
- **Benefit**
  - Enhances feature extraction and model robustness.

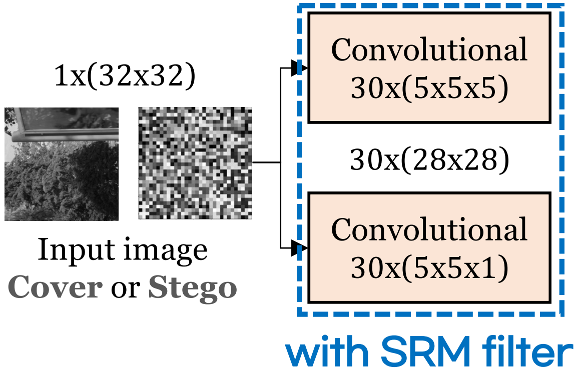
CS-Net: Architecture

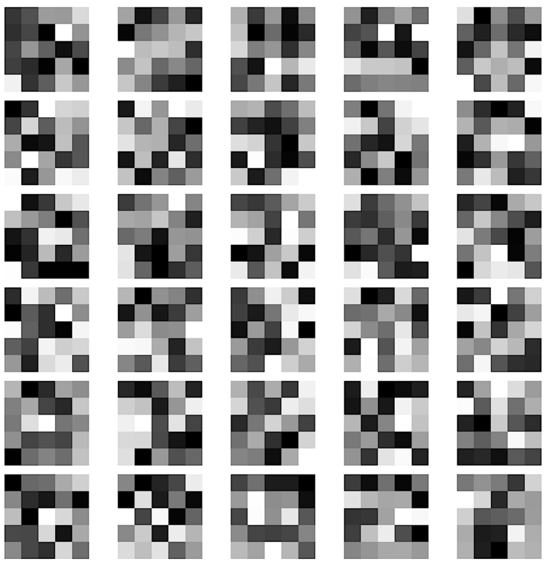


Architecture of CS-Net

- **Preprocessing**
  - It enables effective steganalysis, even for encrypted stego data.
- **Feature Extraction**
  - CNN learn embedding patterns from the stego images.
- **Classification**
  - The last fully connected layers classify images as cover or stego.

Preprocessing : SRM filter

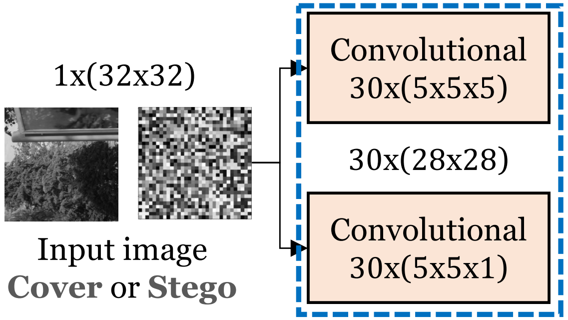




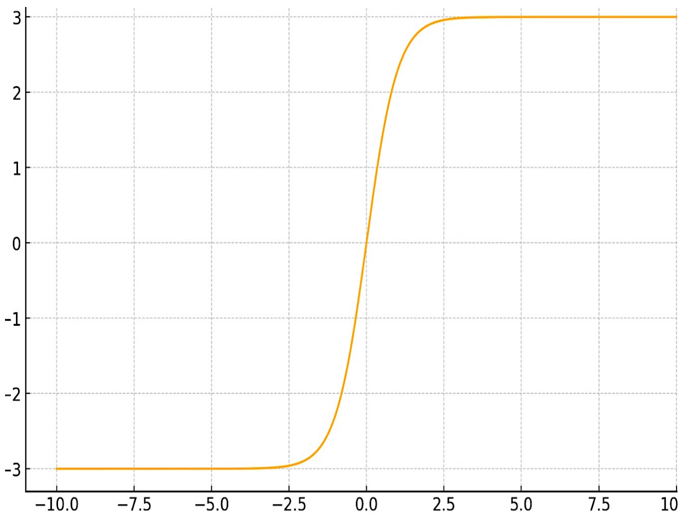
Visualization of 30 SRM filter

- **SRM filter**
  - A type of high-pass filter that extracts image features by considering 30 different directions (30 SRM filter).
  - It is employed as initial filter of Conv2D.
    - Basically, Conv2D layer has a filter (kernel).
    - Thus, 30 SRM filter is used instead of initial weights.
      → **Conv2D** (30,⋯,kernel_initializer = srm_weights, ⋯)
  - It is a general filter for preprocessing in Steganalysis.
    - Ye-Net, Yedroudj-Net and GBRAS-Net used SRM filter in preprocessing.

Preprocessing : Activation

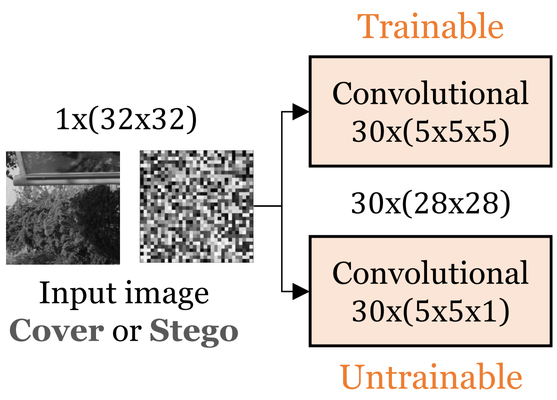


$$3\tanh(x) = 3 \times \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$$



- **3-Tanh activation**
  - Smoothly maps values to the range of -3 to 3, enhancing stability and reducing noise sensitivity in steganalysis.
  - We employed 3Tanh as activation function of Conv2D.
    - Basically, Conv2D layer has an activation.
    - Experimentally identified as the most effective activation function for steganalysis.
  - GBRAS-Net (state-of-the-art) also used 3Tanh for preprocessing.

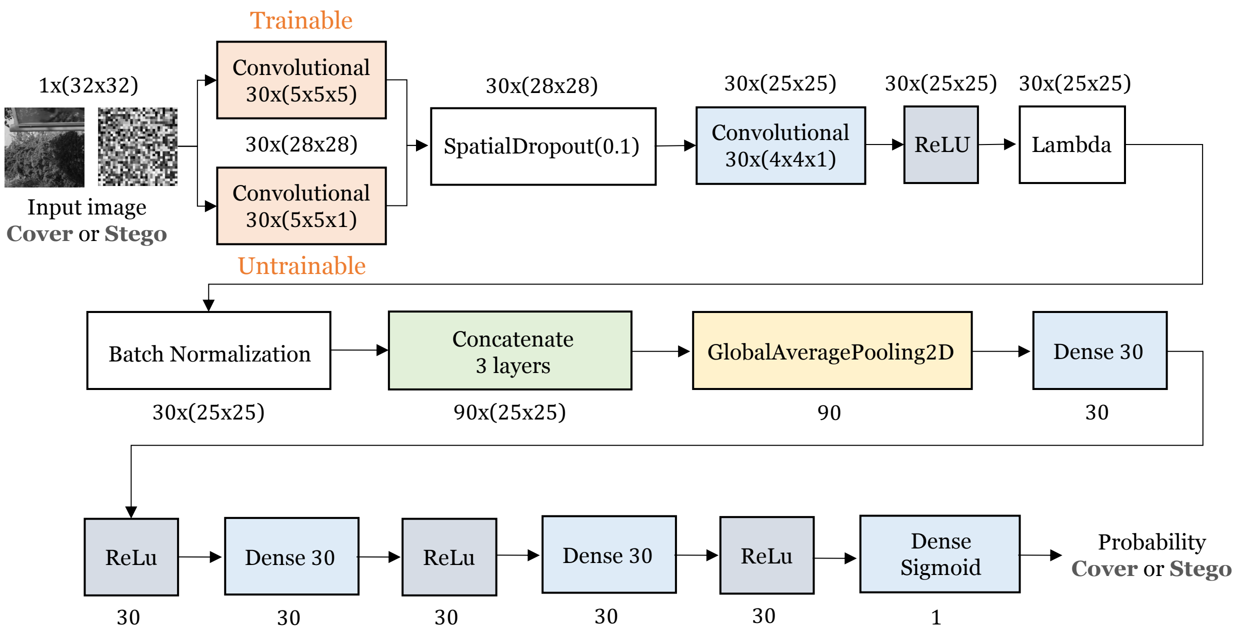
Preprocessing : Architecture



- **Two initial layers**
  - Trainable
    - 30 filters to adaptively learn features during training.
  - Untrainable
    - 30 fixed filters (untrainable weights)

- **Better Training Efficiency**
  - By focusing on pattern refinement, the trainable layers capitalize on the foundation provided by the untrainable filter.
  - It leads to more efficient learning and minimizing overfitting.
  - This synergy ensures faster training convergence while maintaining robust feature extraction.

Neural Network: Architecture



CS-Net's architecture

- Convolutional Layers
  - To train on stego data in image form.
- Average Pooling
  - To retain valuable features, avoiding the information loss.
- Activation functions
  - **3Tanh** (preprocessing layers)
  - **ReLU** (hidden layers)
    - Efficient, **fast and widely used.**
  - **Sigmoid** (output layer)
    - **To distinguish encrypted stego data from cover data**

Neural Network: Architecture

- **Overfitting Prevention**
  - Dropout
    - Randomly drop neurons during training.
  - Batch Normalization
    - Stabilizes learning by normalizing inputs.
  - Residual Connections
    - **Reuses features** across layers, improving stability and reducing overfitting.
  - SGD optimizer
    - Prioritized **stability and reduced overfitting** rather than faster convergence.
    - Ensures better generalization.

Hyperparameters	Descriptions
Epochs	10
Loss function	binary cross-entropy
Optimizer	SGD (learning rate=0.005, momentum=0.95)
Activation function	3-Tanh (Initial), ReLu (Hidden), Sigmoid (Output)
Batch size	32
Parameters	20701 (trainable), 840 (untrainable)
Initial parameters	30 SRM filters and bias

Results

Effects of our Strategies

- **Rotation Strategy for Stego Images**
  - Augments training data by adding 28,000 samples per rotation.
    - Original, 90°, 180°, 270°
  - Increases dataset diversity, reducing overfitting and improving generalization.
  - Enhances feature learning for robust encrypted stego image patterns.

- **Preprocessing with 30 SRM and 3-Tanh**
  - Improves accuracy by 2-8%, even for encrypted stego images.

Performance Analysis for Reduce Rounds

T	E_r	Reduced round (11)		
		Train accuracy	Test accuracy	Reliability
140	0.255	0.7657	0.7512	0.2512
150	0.245	0.7386	0.7201	0.2201
160	0.235	0.7223	0.7192	0.2192
170	0.225	0.7179	0.7028	0.2028
180	0.215	0.6781	0.6702	0.1702
190	0.205	0.6647	0.6531	0.1531
200	0.195	0.6325	0.6208	0.1208
210	0.185	0.6024	0.5974	0.0974
220	0.175	0.5825	0.5734	0.0734

Performance Analysis for Reduce Rounds (11 Rounds)

- T : The number of hidden pixels
- E_r : Embedding rate

$$E_r = \frac{the\ number\ of\ pixels\ with\ values > threshold}{1024\ (32 \times 32)}$$

- **Results**
  - For $E_r \geq 0.175$ **(all cases):**
    Our model can detect stego data (valid).

Reduced-rounds enhance classification reliability, even at lower embedding rates.

Performance Analysis for Full Rounds

T	E_r	Full round (22)		
		Train accuracy	Test accuracy	Reliability
140	0.255	0.6408	0.6364	0.1364
150	0.245	0.6301	0.6298	0.1298
160	0.235	0.5930	0.5872	0.0872
170	0.225	0.5721	0.5692	0.0692
180	0.215	0.5649	0.5669	0.0669
190	0.205	0.5489	0.5487	0.0487
200	0.195	0.5271	0.5295	0.0295
210	0.185	0.5017	0.5016	0.0016
220	0.175	0.5015	0.5012	0.0012

Performance Analysis for Full Rounds (22 Rounds)

- **Result**
  - If accuracy is not exceed 0.51, the steganalysis model is invalid.
  - For $E_r \geq 0.195$: Our model can detect stego data (valid).
  - For $E_r < 0.185$ (lower E_r): The accuracy decreases (invalid).

We can classify at low T (high E_r).

But full-round encryption maintains robust security.

Conclusions

Conclusions

- **Steganalysis for confidentiality through cryptography.**
  - We employ SPECK cipher to generate dataset for secure steganography.
  - It enhances the confidentiality of steganography, making hidden data harder to detect.
- **CS-Net can successfully distinguish stego data from cover data.**
  - We apply rotation strategy and famous preprocessing to improve our performance.
  - We achieve the following results:
    - Round-reduced:
      - Our model can detect stego data for all cases ($E_r \geq 0.175$).
    - Full round:
      - Our model can detect stego data ($E_r \geq 0.195$).
      - The accuracy decreases ($E_r < 0.185$).

Future works

- **Adopting Robust Steganographic Methods**
  - To improve performance of our model, we aim to replace the LSB embedding with more advanced techniques like WOW and J-UNIWARD.

- **Improving Model Generalization**
  - We will develop a robust steganalysis model compatible with diverse cryptographic and steganographic techniques for real-world scenarios.

Thank you for your attention.

E-mail: dudejrdl123@gmail.com