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

https://youtu.be/LIYnccMKYPI

논문

- **보안의 중요성**
  - 미디어 및 통신 기술의 발전으로 인해 디지털 데이터가 빠르게 확장되면서 강력한 보안 대책에 대한 요구가 커짐

- **스테가노그래피 / 암호화**
  - 기존 스테가노그래피 : 추가 암호화 없이 알고리즘을 내장하는 데 의존하는 경우가 많고 스테그아날리시스 기술이 향상됨에 따라 숨겨진 데이터를 탐지하기 쉬워짐

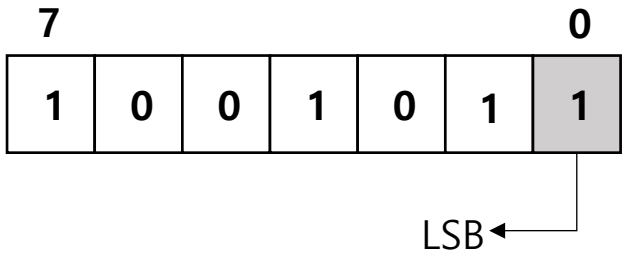
  - 암호화의 한계 : 암호화는 데이터를 효과적으로 암호화하지만, 데이터 자체의 존재를 숨기지는 못함

- **CS-Net**
  - 스테가노그래피(Steganography) 기반 데이터 은닉 : CS – Net은 데이터 은닉을 위해 **LSB(Least Significant Bit) 임배딩 알고리즘**을 사용

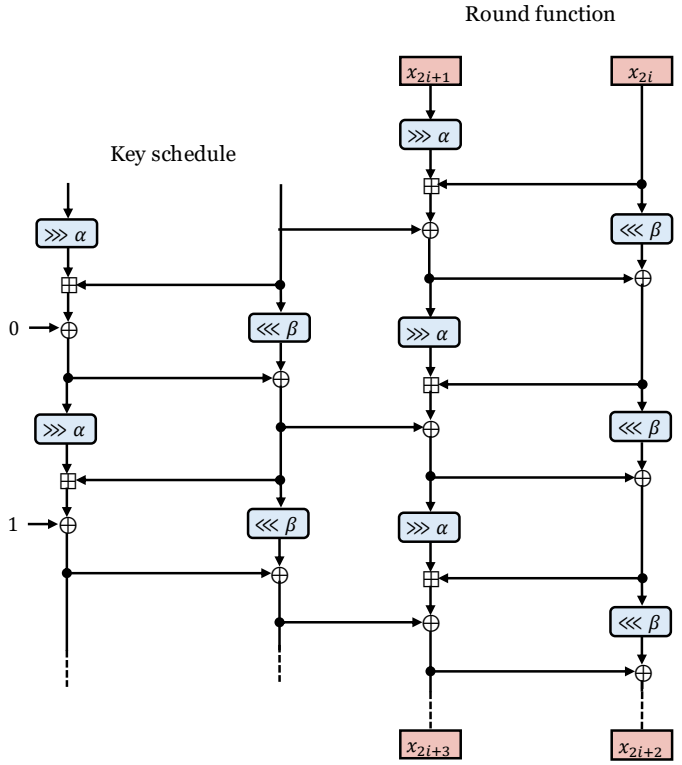
  - 암호화(Crytography) 기반 보안 강화 : CS-Net은 숨겨진 데이터(스테고 이미지)를 추가로 암호화하여 보안을 강화하기 위해 **SPECK 알고리즘**을 사용

LSB / SPECK

- **LSB**
  - LSB 임베딩은 이미지의 가장 낮은 비트에 비밀 정보를 삽입하여 데이터를 숨기는 가장 간단한 기법
- **SPECK**
  - **SPECK**은 NSA가 만든 경량 블록 암호화 알고리즘으로, 보안이 필요한 데이터에 대해 빠르고 효율적인 암호화를 제공
  - 우리의 프레임워크에서는 SPECK-32/64 버전을 사용



LSB Steganography



Schematic of SPECK encryption.

Comparison steganography

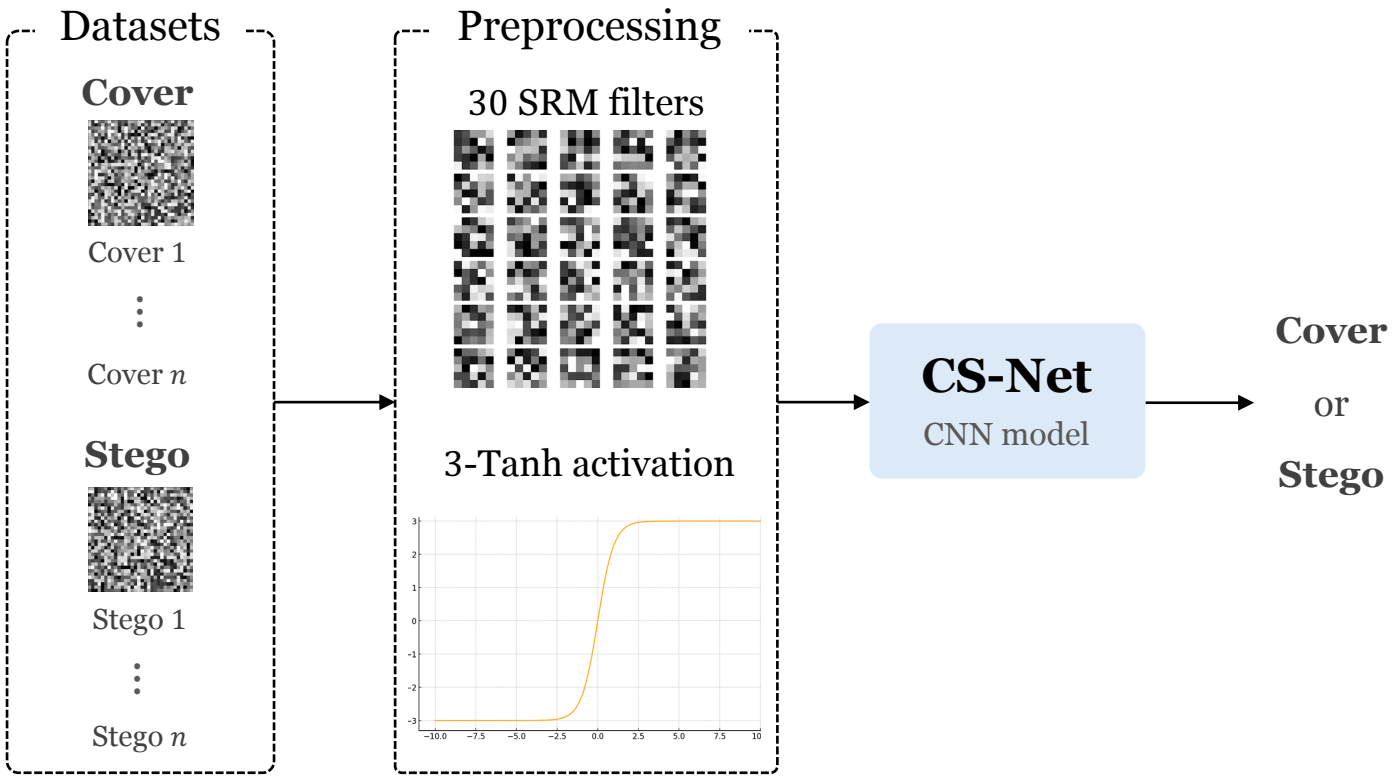
- **기존의 스테가노그래피 기술**
  - 기존의 스테가노그래피 기술은 주로 비밀 데이터를 암호화하지 않고 임베딩 알고리즘에 의존해 왔기 때문에 보안이 이러한 알고리즘의 성능에 크게 의존함
  - 이러한 의존성은 스테그아날리시스 기술이 향상되면서 쉽게 탐지될 수 있게 됨
  - 최근 스테고 이미지의 탐지 정확도를 높이기 위해 합성곱 신경망(CNN)을 활용한 딥러닝 기반 많은 스테그아날리시스 기법이 제안됨
  - CNN모델은 데이터에서 고차원 특징을 자동으로 학습하도록 설계되어 기존의 통계 분석 방법보다 상당한 개선을 제공
  - 이처럼 스테가노그래피와 스테그아날리시스 기술들을 통합하여 기밀성을 향상시키려는 수많은 제안에도 스테가노그래피와 암호 알고리즘을 결합한 프레임 워크는 없었음

Table 1: Comparison with related works for steganalysis.

Framework	Steganography	Cryptography	Descriptions
Xu-Net [6]	✓ (WOW, S-UNIWARD)	✗	HPF and Tanh activation
Ye-Net [7]			SRM filter and TLU activation
Yedroudj-Net [8]			Combination of Xu-Net and Ye-Net
GBRAS-Net [9]			High accuracy and fewer parameters
CS-Net (Ours)	✓ (LSB)	✓ (SPECK)	Rotation and preprocessing strategy

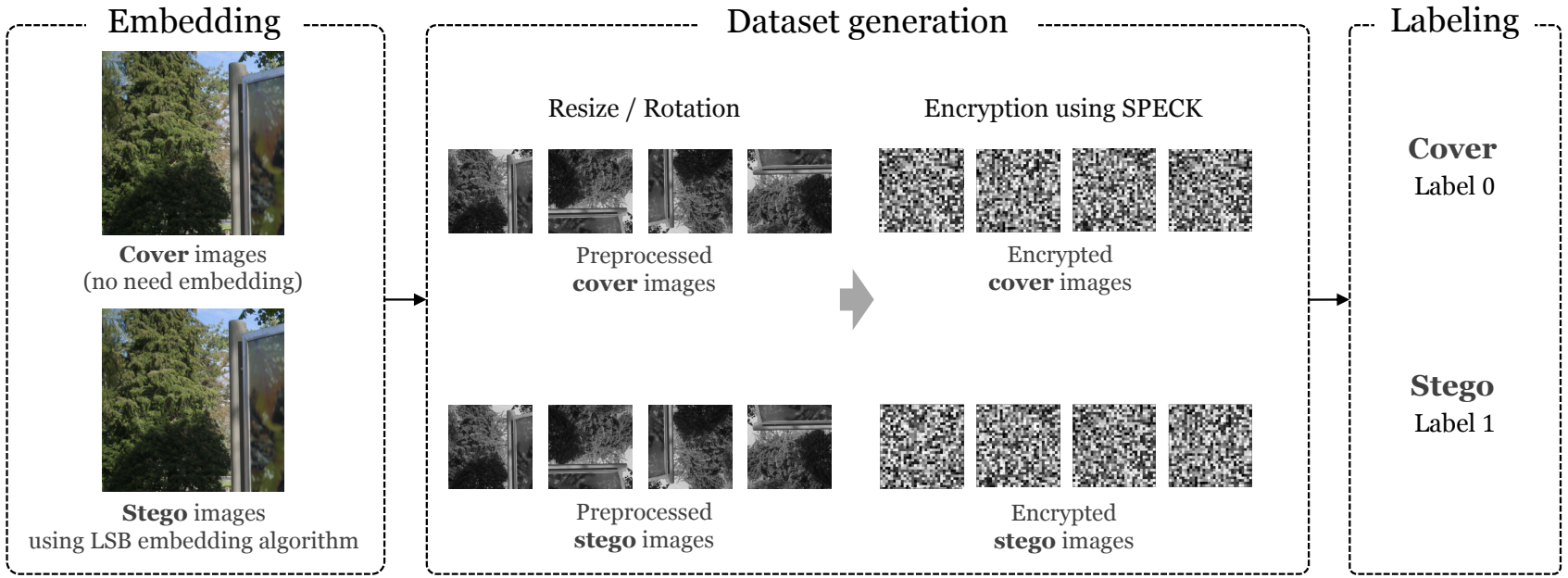
Cs-Net

- **Cs-Net 구조**
  - CS-Net은 데이터 세트 구성, 전처리 기법, 신경망 아키텍처 등 다양한 기술을 통합하여, 레이블이 지정되지 않은 암호문 데이터가 커버 이미지인지 스테고 이미지인지 판별



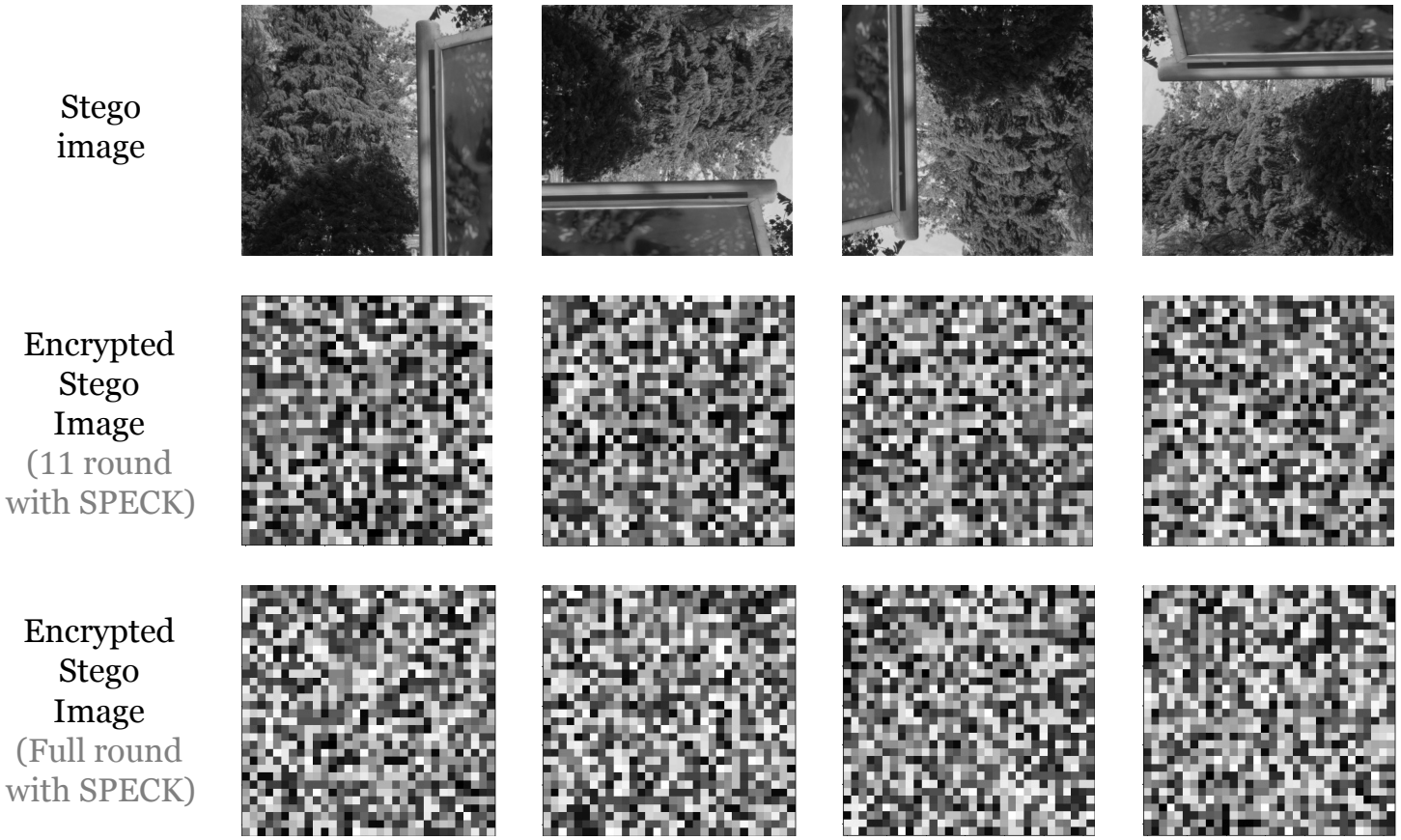
Dataset

- **데이터셋 생성 과정**
  - 실험 환경을 고려하여 커버 및 스테고 이미지를 모두 32x32 gray 이미지로 변환
  - 변환 된 이미지는 LSB 임베딩 기술을 통하여 28,000장의 스테고 이미지를 생성
  - 이후 로테이션 기술을 사용하여 90, 180, 270, 360도의 112,000개(=28,000x4) 이미지로 증가
- **암호화 과정**
  - 생성된 스테고 이미지를 암호화하기 위해 SPECK(32/64) 알고리즘 사용
  - SPECK-32/64의 풀라운드와 축소 라운드(22, 11라운드) 진행



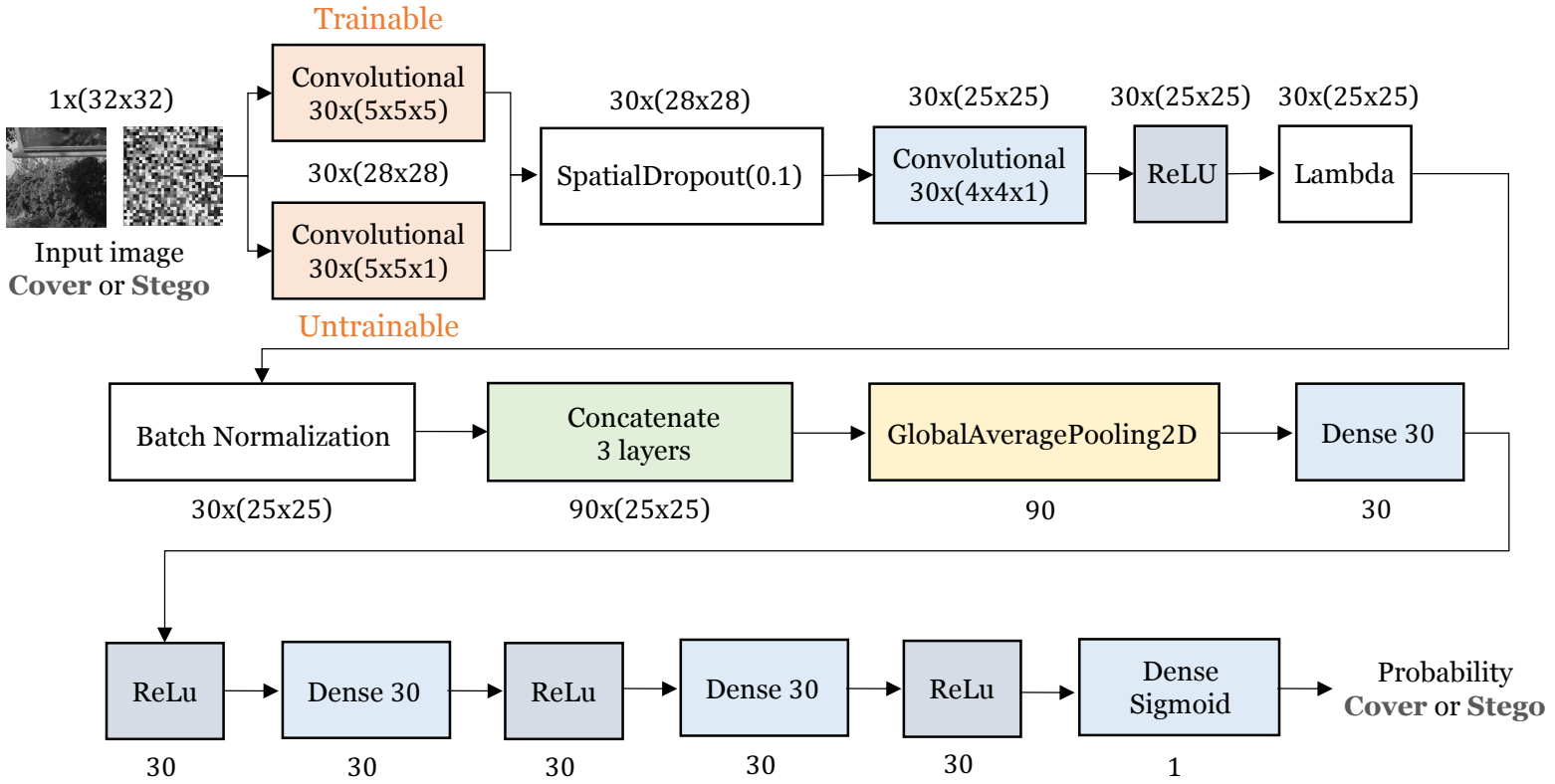
Novel Rotation Strategy

- **Rotation Stategy**
  - 각 스테고 이미지를 90, 180, 270도 회전하여 훈련 데이터 셋을 증강함
  - 회전된 스테고 이미지를 사용하여 여러 방향에서 잔여 정보를 추출함
  - 이로 인해 더 다양한 범위의 입력을 제공하여 모델이 학습하는 데 도움
  - 우리의 모델에서 다양한 암호화 방향에 해당하는 패턴을 효과적으로 학습함



Network Architecture

- **Network Architecture**
  - 네트워크는 두 개의 초기 합성곱 계층으로 시작
  - 첫 번째 계층은 학습할 수 없는 가중치를 사용하는 반면 두번째 계층은 학습 가능한 가중치를 사용(4x4 커널 사용)
  - 이 기술을 사용하면 모델이 사전 정의된 필터와 적응 학습의 이점을 모두 얻을 수 있음
  - 과적합을 방지하기 위해 드롭 아웃, 잔여 네트워크, 배치 정규화를 적용 최적화에는 일반화를 위한 SGD를 사용
  - 마지막으로 평균 풀링을 사용해 데이터의 중요한 정보를 유지하며, 마지막 계층에서 시그모이드 활성화 함수를 통해 이미지가 커버 또는 스테고인지 분류



Hyperparameters of CS-Net.

- **Hyperparameters of Cs-Net**
  - Epochs = 10
  - Loss function = binary cross - entropy
  - SGD = learning rate = 0.005 / momentum = 0.95
  - Activation function = 3– Tanh, ReLu, Sigmod
  - Batch size = 32
  - Parameters = 20701 / 840

Table 2: Hyperparameters of CS-Net.

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

- **Cs-Net 성능결과**
  - 모델은 Speck 32/64 암호화 알고리즘의 전체 라운드**(22라운드) / (11라운드)**에서 테스트 됨
  - **T는 임계값, Er은 임베딩 비율을** 의미
  - 학습의 정확도, 테스트 정확도, 신뢰성에 대한 결과가 포함

Table 3: Result of CS-Net.

T	E_r	Full round (22)			Reduced round (11)		
		Train accuracy	Test accuracy	Reliability	Train accuracy	Test accuracy	Reliability
140	0.255	0.6408	0.6364	0.1364	0.7657	0.7512	0.2512
150	0.245	0.6301	0.6298	0.1298	0.7386	0.7201	0.2201
160	0.235	0.5930	0.5872	0.0872	0.7223	0.7192	0.2192
170	0.225	0.5721	0.5692	0.0692	0.7179	0.7028	0.2028
180	0.215	0.5649	0.5669	0.0669	0.6781	0.6702	0.1702
190	0.205	0.5489	0.5487	0.0487	0.6647	0.6531	0.1531
200	0.195	0.5271	0.5295	0.0295	0.6325	0.6208	0.1208
210	0.185	0.5017	0.5016	0.0016	0.6024	0.5974	0.0974
220	0.175	0.5015	0.5012	0.0012	0.5825	0.5734	0.0734

감사합니다.