ANTAGONISTIC WATERMARKING TO ATTACK DEEP NEURAL NETWORKS

Gauri Bhosale[1], **Prof. Sharad Rokade**[2]

[1]*PG Student, Computer Engineering Dept. of SVIT Nashik Maharashtra, India*
[2]*Assosiate Professor Computer Engineering Dept. of SVIT Nashik Maharashtra, India*

---***---

Abstract -*Training machine learning (ML) models is expensive in terms of computational power, amounts of labeled data and human expertise .Thus, ML models constitute intellectual property (IP) and business value for their owners. Existing watermarking schemes are ineective against IP theft via model extraction since it is the adversary who trains the surrogate model. Specially, the watermark is adjusted iteratively on location, transparency, color, angle and size which are determined by only 9 parameters. We dene two types of attack to better simulate the watermark approaches in reality, respectively the watermark is constrained in either transparency or size. V3 model with high success rates, but also transferable to other models with high condence, such as the Recognition developed by Amazon. In this paper, we introduce Adversarial Watermarking of Neural Networks, the rst approach to use watermarking to deter model extraction IP theft. This set is a watermark that will be embedded in case a client uses its queries to train a surrogate model. We show that DAWN is resilient against two state-of-the-art model extraction attack..*

Key Words: **Adversarial attack, Watermark, Deep Neural Networks.**

1. INTRODUCTION

•Deep neural networks have made tremendous progress in the area of multimedia representation, training neural models requires a large amount of data and time.

•Introduction would effect the understanding of deep learning models remains unstudied. In this work, we propose a visible adversarial attack method that transforms and places a provided watermark on the target image to in-terfere the classification result from an Inception V3 model ,which is pretrained on ImageNet. Specifically, the watermark is adjusted iteratively on location, transparency, color, angle and size which are determined by only 9 parameters. We de-fine two types of attack to better simulate the watermark approaches in reality, respectively the watermark is constrained in either transparency or size. Thirdly, we propose a discovery framework for the private data chain and demonstrate its feasibility and effectiveness by experiments. Which provide a reference to develop a system software assuring the safety for personal privacy data in big data.

Our solution, in turn, reduces the non-recurring engineering cost and enables model designers to incorporate specific Watermark (WM) information during the training of a neural network with minimal changes in their source code and overall training overhead. By introducing DeepSigns, this paper makes the following contributions:

• **Enabling effective IP protection for DNNs.** A novel watermarking methodology is introduced to encode the pdf of activation maps and effectively trace the IP ownership.

• **Characterizing the requirements for an effective watermark embedding in the context of deep learning.** . We provide a comprehensive set of metrics that enables quantitative and qualitative comparison of current and pending DNN-specific IP protection methods.

• **Devising a careful resource management and accompanying API.** A user-friendly API is devised to minimize the non-recurring engineering cost and facilitate the adoption of DeepSigns within contemporary DL frameworks including TensorFlow, Pytorch, and Theano.

• **Analysis of various DNN topologies.** Through extensive proof-of-concept evaluations, we investigate the effectiveness of the proposed framework and corroborate the necessity of such solution to protect the IP of an arbitrary DNN and establish the ownership of the model designer. This paper opens a new axis for the growing research in secure deep learning. This work sheds light on previously unexplored impacts of IP protection on DNNs' performance. DeepSigns provides a development tool for the research community to better protect their innovative DNN designs. Our tool is open source and will be publicly available

System Architecture

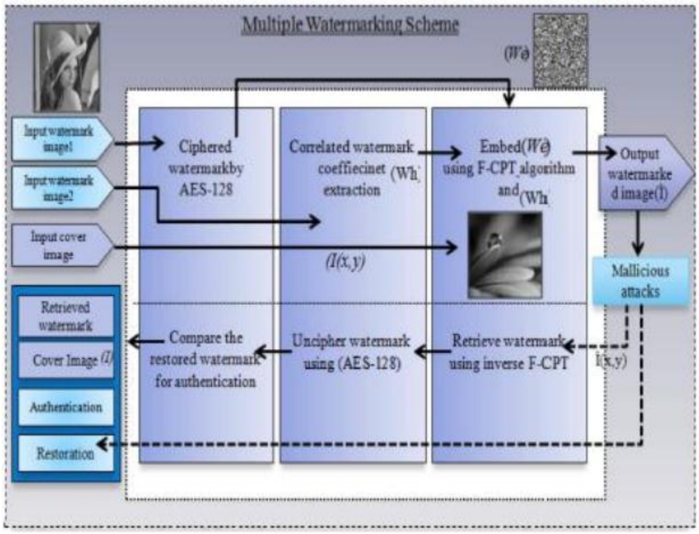
Multiple Watermarking Scheme
(We)
Input watermark image1
Input watermark image2
Input cover image
Ciphered watermarkby AES-128
Correlated watermark coeffiecinet (Wh extraction
Embed(We) using F-CPT algorithm and (Wh
Output watermarked image(I)
(I(x,y)
Mallicious attacks
Retrieved watermark
Cover Image (I)
Authentication
Restoration
Compare the restored watermark for authentication
Uncipher watermark using (AES-128)
Retrieve watermark using inverse F-CPT
I(x,y)

Fig -1: System architecture of proposed solution

System architecture is the conceptual design that defines the structure and behavior of a system. An design explanation is a prescribed description of a system, systematized in a way that supports perceptive about the fundamental properties of the system. It outlines the system modules or building blocks and delivers a plan from which products can be secured, and systems developed, that will work organized to implement the whole system. The watermark extraction process extracts watermark from watermarked image and it is exactly reverse process as that of watermark embedding process. In case of non-blind watermarking both cover image and watermarked image are required during watermark extraction process, while in blind image watermarking only watermarked image is required in watermark extraction process. The variety of attacks are applied such as noise addition, noise filtering, rotation, scaling, translation, Gamma correction, resizing, cropping, compression. These attacks are considered to evaluate the robustness of developed image watermarking techniques under proposed system. The proposed system also includes MEO based grey scale image watermarking techniques which is proposed for optimization of perceptual quality and robustness under high payload scenario.

DeepSigns Overview

Demonstrates the global flow of DeepSigns framework. DeepSigns consists of two main phases: watermark embedding and watermark extraction. The watermarked DNN can be employed as a service by third-party users either in a white-box or a black-box setting depending on whether the model internals are transparent to the public or not. DeepSigns is the first DNN watermarking framework that is applicable to both white-box and black-box security models.

Watermark Embedding

DeepSigns takes the DNN architecture and the owner-specific watermark signature as its input. The WM signature is a set of arbitrary binary strings that should be generated such that each bit is independently and identically distributed (i.i.d.).Then, the underlying DNN is trained (fine-tuned) such that the ownerspecific WM signature is encoded in the pdf distribution of activation maps obtained at different DNN layers. Model distribution is a common approach in the machine learning field (e.g., the Model Zoo by Caffe Developers, and Alexa Skills by Amazon). Note that even though models are voluntarily shared, it is importantto protect the IP and preserve copyright of the original owner.

Watermark Extraction

To verify the IP of a remote DNN and detect potential IP infringement, the model owner first needs to query the remote DNN service with WM keys generated in the WM embedding phase and obtain the corresponding activation maps. DeepSigns then extracts the WM signature from the pdf distribution of the acquired activation maps. It next computes the Bit Error Rate (BER) between the extracted signature in each layer and the corresponding true signature. If the BER at any layer is zero, it implies that the owner's IP is deployed in the remote DNN service.

2. General system requirement

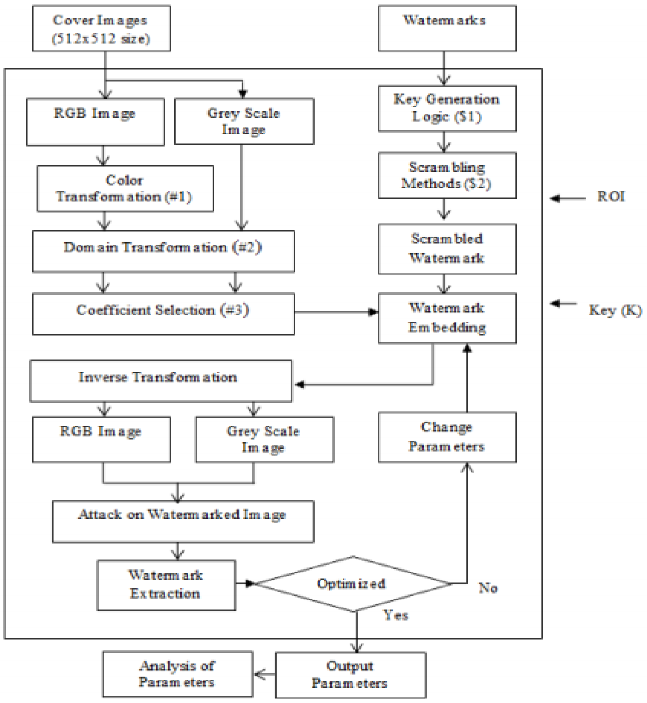

Fig -2: Block Diagram

Hardware Requirements

- Processor: - i3
- RAM: - 2 GB.
- Hard Disk: - 500 GB. (As per data)

Software Requirements

- Operating System: Windows XP, Windows 7.
- Internet Browser: Google Chrome, Mozilla.
- Front End: JAVA
- Back End: SQL.
- Database: MySQL Server 2008
- Software: JAVA

Algorithm

```
Encrypted Watermark [13]
AES-128, key size=128
One round of AES consists of:
{ Byte substitution

{ Permutation

{ Arithmetic operation

{ XOR with generated key
{ Part image in 4 blocks of 128 128 bits
For block b=1:4 Permutation
P =(Input (Arithmetic Operation) XOR) / Byte substitution Input
(Row i=1, 2,3,..128) to AES-128 End.
{Encrypted watermark achieved.
/* obtain watermark */
$x \leftarrow Tr(Ii)$; $Loss(Ii) \leftarrow -R(x)$;
 /* perform one step update */
$g \leftarrow \nabla Loss(Ii)$; $Ii+1 \leftarrow Ii - \eta g$;
if performing SSIM heatmap attenuation then
hssim ← SSIMheatmap(Io, Ii+1);
 /* attenuate watermark with psycho-visual heatmap */
 $I\delta \leftarrow Ii+1 - Io$; $Ii+1 \leftarrow Io + hssim \otimes I\delta$;
 end /* compute indicator values */
Compute Dist(Io, Ii+1) $i \leftarrow i + 1$;
 until stopping criterion met; $Iw \leftarrow Ii$ ;
Terminate
```

4. CONCLUSION

In this paper, we proposed a visible adversarial attack approach utilizing watermarks, with two types of attack to simulate the real-world cases of watermarks and have successfully interfered the judgment from some state-of-the-art deep learning models. Moreover, partial adversarial samples show great transferability onto other models including the Recognition. In conclusion, we believe this work suggests that the robustness of current object recognition models are yet to be further improved, and more defense approaches shall be employed.

REFERENCES

[1] Y. Nagai, Y. Uchida, S. Sakazawa, and S. Satoh, Digital watermarking for deep neural networks, International Journal of Multimedia Information Retrieval,vol. 7, no. 1, 2018.

[2] "Embedding Watermarks into Deep Neural Networks",https://arxiv.org/abs/1701.04082

[3] E. L. Merrer, P. Perez, and G. Tredan, Adversarial frontier stitching for remoteneural network watermarking, arXiv preprint arXiv:1711.01894),2017.

[4] Y. Adi, C. Baum, M. Cisse, B. Pinkas, and J. Keshet, Turning your weakness into a strength: Watermarking deep neural networks bybackdooring, Security Symposium 2018.

[5] K. Grosse, P. Manoharan, N. Papernot, M. Backes, and P. McDaniel,On the (statistical) detection of adversarial examples, arXiv preprint arXiv:1702.062802017.

[6] Y.Chen, C. Qiao and X. Yu, Convolutional neural networks for medical image analysis: Full training or ne tuning? IEEE transactions on medical imaging, vol. 35, no. 5, 2018.

[7] B. D. Rouhani, M. Samragh, T. Javidi, and F. Koushanfar, Safe machinelearning and defeat-ing adversarial attacks, IEEE Security and PrivacyMarch 2018.