Reproducibilty of Boosting Adversarial Transferability via Gradient Relevance Attack

Anonymous authors
Paper under double-blind review

Abstract

This paper presents a reproducibility study of "Boosting Adversarial Transferability via Gradient Relevance Attack" by Zhu et al., a paper that introduces the Gradient Relevance Attack (GRA) method. GRA enhances the transferability of adversarial examples across different machine learning models, improving black-box adversarial attacks. We successfully replicated the key experiments, focusing on the gradient relevance framework and the decay indicator. Our methodology involved reimplementing the GRA algorithm and evaluating it on the same set of models used in the original paper. We achieved attack success rates comparable to those of the original article, within a margin of 1%, confirming the effectiveness of the GRA method. Additionally, we extended the original work by introducing a dynamic learning rate (α) that adjusts the step size based on the cosine similarity between the current momentum and the average gradient. An adjustment factor (γ) of 0.01, with thresholds of 0.75 and 0.25, modulates the step size. Our findings suggest that this adaptive step size mechanism can lead to faster convergence and potentially improved attack performance in certain scenarios. This study validates the GRA method and explores avenues for further improving adversarial transferability through dynamic parameter adjustments.

github link : https://anonymous.4open.science/r/MLRC-6E15

1 Introduction

Deep Neural Networks (DNNs) have revolutionized computer vision through unprecedented performance on tasks ranging from image classification to medical diagnosis He et al. (2016); Huang et al. (2017); Girshick (2015); Bojarski et al. (2016); Taigman et al. (2014); Chen et al. (2018); Wang et al. (2023). However, their susceptibility to adversarial examples—inputs modified with imperceptible perturbations that induce misclassification —exposes critical security vulnerabilities Athalye et al. (2018); Goodfellow et al. (2014); Szegedy et al. (2013); Carlini & Wagner (2017) . While white-box attacks achieve near-perfect success rates with full model access, the practical black-box scenario— where attackers must transfer adversarial examples between models— remains challenging, particularly against defense-enhanced systems.

The paper "Boosting Adversarial Transferability via Gradient Relevance Attack" Hegui Zhu (2023) introduced a novel approach, the Gradient Relevance Attack (GRA), to enhance the transferability of adversarial examples in black-box settings. GRA leverages a gradient relevance framework and a decay indicator to improve the effectiveness of adversarial attacks.

The Transferability Challenge

The core challenge lies in adversarial transferability—the ability of perturbations crafted on one model (source) to deceive other architecturally distinct models (targets). Traditional gradient-based methods like + suffer from:

- **Gradient misalignment:** Update directions optimized for source models poorly generalize to targets.

- **Oscillation effects:** Fixed step sizes cause perturbation sign fluctuations (Fig. 2-3 of Hegui Zhu (2023)).
- **Defense vulnerability:** Poor performance against adversarially trained models and input transformations.

Gradient Relevance Framework

Zhu et al. Hegui Zhu (2023) address these limitations through two key mechanisms:

Gradient Relevance Framework

- **Attention-inspired weighting:** Treats the current gradient as a "query" and neighborhood gradients as "keys" using cosine similarity (Fig. 4 of Hegui Zhu (2023)).
- **Adaptive correction:** Blends gradients via:

  $$WG_t = s_t \cdot G_t + (1 - s_t) \cdot \overline{G_t} \tag{1}$$

  where s_t measures alignment between the current gradient G_t and neighborhood average $\overline{G_t}$.

Decay Indicator

- **Oscillation mitigation:** Dynamically adjusts step size via:

  $$M_{t+1} = M_t \odot \left(M^e_{t+1} + \eta \cdot M^d_{t+1} \right) \tag{2}$$

- Attenuation factor $\eta = 0.94$ reduces perturbation magnitude at sign-flip pixels.

In this work, we present a reproducibility study of the GRA method. We successfully replicated the key experiments and results of the original paper, achieving comparable attack success rates. Furthermore, we extend the original work by introducing a dynamic learning rate that adapts the step size on the basis of the cosine similarity between the current momentum and the average gradient. This dynamic learning rate, modulated by an adjustment factor and thresholds, aims to improve convergence and attack performance. Our findings validate the effectiveness of the GRA method and demonstrate that the introduction of a dynamic learning rate has the potential to further enhance adversarial transferability.

The rest of this paper is structured as follows: Section 2 provides a brief overview of the related work on adversarial attacks and defenses. Section 3 describes our reproduction of the GRA method, including implementation details and experimental setup. Section 4 presents our extension to the GRA method with the dynamic learning rate. Section 5 discusses the experimental results, including a comparison with the findings of the original article and an analysis of the impact of the dynamic learning rate. Finally, Section 6 concludes the paper and outlines potential directions for future research.

2 Scope of Reproducibility

This paper presents a rigorous reproducibility study of "Boosting Adversarial Transferability via Gradient Relevance Attack" by Zhu et al. (2023). Our primary goal is to validate the key claims and experimental results presented in the original work, while also exploring potential improvements through a novel extension. Specifically, we focus on reproducing the following aspects:

Comparative Attack Performance:

- GRA achieves higher attack success rates (ASR) than VTMI-FGSM, VTNI-FGSM, and Admix across diverse models.

- GRA remains effective against both normally and adversarially trained models.

Impact of Combined Transformations:

- Combining GRA with DI, TI, and SI enhances transferability, outperforming standalone GRA.

Ablation Study:

- Gradient relevance framework and decay indicator significantly impact attack performance.
- Varying hyperparameters (sample quantity m, upper bound factor β, and attenuation factor η) affects results.

Extension: Dynamic Learning Rate Adaptation:

- Introducing an adaptive step size (α) using cosine similarity improves convergence.
- Varying adjustment factors (γ) and similarity thresholds affects performance.
- Extended GRA shows superior transferability over the original method.

This study validates GRA's robustness and explores dynamic parameter tuning for improved adversarial transferability.

3 Methodology

3.1 Description of Adversarial Attacks

This section describes the proposed Gradient Relevance Attack (GRA) and its enhancement algorithms.

3.1.1 Gradient Relevance Attack (GRA)

The Gradient Relevance Attack (GRA), proposed by Zhu et al. (2023), improves adversarial transferability by leveraging two key mechanisms: the **Gradient Relevance Framework** and the **Decay Indicator**. The Gradient Relevance Framework adaptively combines the current gradient with neighborhood gradients using cosine similarity to stabilize updates. The Decay Indicator dynamically adjusts the step size for each pixel based on changes in gradient direction, mitigating oscillations during optimization.

Algorithm 1 Original Gradient Relevance Attack (GRA)

```
Input: Source model $F_\psi$, clean image $x_{clean}$, true label $y_{true}$, iterations $T$, momentum decay $\mu$, attenuation factor $\eta$, neighborhood samples $m$, noise bound $\beta\varepsilon$
Output: Adversarial example $x_T^{adv}$
Initialize $\alpha = \epsilon/T$, $g_0 = 0$, $M_0 = 1/\eta$, $x_0^{adv} = x_{clean}$
for $t = 0$ to $T - 1$ do
    Compute current gradient: $G_t(x) = \nabla_{x_t^{adv}} \mathcal{L}(x_t^{adv}, y_{true})$
    Generate $m$ neighbor samples $x_t^i = x_t^{adv} + \gamma_t^i$ where $\gamma_t^i \sim \mathcal{U}(-(\beta\varepsilon)^d, (\beta\varepsilon)^d)$
    Compute average neighborhood gradient: $\bar{G}_t(x) = \frac{1}{m}\sum_{i=1}^{m} \nabla_{x_t^i} \mathcal{L}(x_t^i, y_{true})$
    Compute cosine similarity $s_t = \frac{G_t(x) \cdot \bar{G}_t(x)}{\|G_t(x)\|_2 \|\bar{G}_t(x)\|_2}$
    Compute weighted gradient: $WG_t = s_t \cdot G_t + (1 - s_t) \cdot \bar{G}_t$
    Update momentum: $g_{t+1} = \mu \cdot g_t + \frac{WG_t}{\|WG_t\|_1}$
    Update decay indicator: $M_{t+1} = M_t \odot (M_{t+1}^e + \eta \cdot M_{t+1}^d)$
    Update adversarial example: $x_{t+1}^{adv} = \text{Clip}\{x_t^{adv} + \alpha \cdot M_{t+1} \odot \text{sign}(g_{t+1})\}$
end for
```

3.1.2 Extended GRA

The extended GRA algorithm introduces an adaptive learning rate mechanism that dynamically adjusts α based on gradient alignment stability. Fixed thresholds τ_{high} and τ_{low} with an adjustment factor γ help fine-tune the update step based on gradient relevance. Exponential clipping keeps α within a stable range, allowing larger updates when the adversarial example is far from optimal and smaller updates as convergence nears. The theoretical basis links gradient stability to optimal step size via:

$$\alpha_{t+1} = \alpha_t(1 + \gamma \cdot \text{sign}(\bar{s}_t - \tau^*))$$

where τ^* represents the optimal similarity threshold. Our fixed threshold approach approximates this ideal case while maintaining computational efficiency.

Algorithm 2 Extended Gradient Relevance Attack (GRA)

```
Input: Source model $F_\psi$, clean image $x_{clean}$, true label $y_{true}$, iterations $T$, momentum decay $\mu$, attenuation factor $\eta$, neighborhood samples $m$, noise bound $\beta\varepsilon$
Output: Adversarial example $x_T^{adv}$
Initialize $\alpha = \epsilon/T$, $g_0 = 0$, $M_0 = 1/\eta$, $x_0^{adv} = x_{clean}$ Additional parameters: Adjustment factor $\gamma$, fixed thresholds $\tau_{high}$, $\tau_{low}$
for $t = 0$ to $T - 1$ do
    Compute $G_t(x)$ and $\bar{G}_t(x)$ as in original GRA
    Compute cosine similarity $s_t$ and weighted gradient $WG_t$
    Compute mean similarity: $\bar{s}_t = \mathbb{E}[s_t] = \frac{1}{m}\sum_{i=1}^{m} s_t^i$
    Adjust learning rate dynamically:
        $\alpha_{t+1} = \begin{cases} \alpha_t(1+\gamma) & \bar{s}_t > \tau_{high} \\ \alpha_t(1-\gamma) & \bar{s}_t < \tau_{low} \\ \alpha_t & \text{otherwise} \end{cases}$
    Update momentum $g_{t+1}$ and decay indicator $M_{t+1}$ as in original GRA
    Apply dynamic learning rate: $x_{t+1}^{adv} = \text{Clip}\{x_t^{adv} + \alpha_{t+1} \cdot M_{t+1} \odot \text{sign}(g_{t+1})\}$
end for
```

3.2 Datasets and Models

3.2.1 Datasets

The experimental framework utilizes a standardized evaluation protocol based on the ILSVRC 2012 validation set ILSVRC2012, following the established methodology from the original GRA paper 1,GRA2. From the 50,000-image validation set, we select 1,000 high-confidence samples where all evaluated models achieve $\geq 99\%$ classification accuracy under clean conditions. This curation ensures meaningful measurement of adversarial perturbation effectiveness against robust baselines.

3.2.2 Model Architectures

Experiments employ four standard source models and seven target models following the original paper's configuration[1][2]:

Source Models: 1. Inception-v3 (Inc-v3)[30]: 27M parameters 2. Inception-v4 (Inc-v4)[29]: 42M parameters 3. Inception-ResNet-v2 (IncRes-v2)[29]: 55M parameters 4. ResNet-v2-101 (Res-101)[13]: 44M parameters

Target Models: - Standard classifiers: Inc-v3, Inc-v4, IncRes-v2, Res-101 - Adversarially trained variants: - adv-Inception-v3 (Inc-v3adv)[33] - ens3-adv-Inception-v3 (Inc-v3ens3) - ens4-adv-Inception-v3 (Inc-v3ens4) - ens-adv-Inception-ResNet-v2 (IncRes-v2ens)

Model	Attack	Inc-v3	Inc-v4	IncRes-v2	Res-101	Inc-v3$_{ens3}$	Inc-v3$_{ens4}$	IncRes-v2$_{ens}$	Average
Inc-v3	VTMI	100.0*	72.1	69.4	61.5	33.7	30.1	17.2	54.9
	VTNI	100.0*	75.3	74.1	66.2	34.3	31.4	19.4	57.3
	Admix	100.0*	81.5	79.7	74.1	41.2	38.5	20.4	62.2
	GRA	99.8*	86.8	85.2	78.6	57.9	55.9	41.2	71.9
Inc-v4	VTMI	77.9	99.8*	70.2	63.3	38.4	37.2	24.5	58.8
	VTNI	83.5	99.9*	76.3	66.1	40.5	39.0	23.9	61.3
	Admix	87.9	99.5*	83.0	78.4	55.1	50.9	33.2	69.8
	GRA	89.2	99.1*	86.4	79.2	66.2	62.8	50.3	76.2
IncRes-v2	VTMI	77.6	72.3	98.0*	66.9	46.9	40.5	34.1	62.3
	VTNI	80.5	76.2	98.1*	69.2	48.2	42.1	33.0	64.0
	Admix	89.3	87.1	99.0*	81.4	65.7	55.9	49.8	75.7
	GRA	86.2	83.3	97.1*	79.6	68.9	61.1	56.3	76.1
Res-101	VTMI	74.9	67.8	70.1	99.3*	45.1	40.0	29.3	60.9
	VTNI	78.5	74.1	72.8	99.5*	47.9	41.3	30.9	63.6
	Admix	85.9	81.2	80.5	99.8*	51.8	44.2	34.2	67.9
	GRA	87.4	83.5	84.1	99.6*	72.1	67.5	57.4	78.8

Table 1: The attack success rates (%) on seven models by a single attack. The adversarial examples are generated on Inc-v3, Inc-v4, IncRes-v2, and Res-101 separately. * denotes the success rate of the white-box attack and the result in bold is the best.

Defended Models: 1. Pixel Deflection (PD)[26] + ResNet-v2-50 2. Neural Representation Purifier (NRP)[24] + Inc-v3ens3 3. JPEG Compression[12] + Inc-v3ens3 4. ComDefend[15] + Inc-v3ens3 5. Feature Distillation (FD)[21] + Inc-v3ens3

All models maintain original training configurations from their respective sources[1][2]. For adversarial variants, we utilize publicly released weights from Madry et al.'s ensemble adversarial training framework[33]. Defense mechanisms are implemented as preprocessing modules without retraining base classifiers.

3.3 Experimental Setup

The experimental setup largely mirrors the original paper's methodology [1] to ensure a high degree of comparability while addressing resource limitations. Deviations are explicitly outlined below:

All experiments were conducted using **Google Colab**(T4 GPU) and **Kaggle**(NVIDIA Tesla P100) environments. Key attack parameters were kept consistent with the original paper [1] (L_∞ perturbation budget $\epsilon = 16$, iteration count $T = 10$) to isolate the impact of our dynamic learning rate adaptation.

Base Implementation: Core GRA algorithm implemented using original authors' codebase from `https://github.com/ryc-98/gra` [1]

Practical Online Systems: Evaluation on Tencent Cloud and Baidu AI Cloud APIs was omitted due to resource and API access constraints. While these practical systems are valuable for real-world assessment, our primary focus is on evaluating the core improvement in adversarial transferability achieved through our proposed dynamic learning rate adaptation.

The primary focus of the experimental evaluation remains on comparing the transferability of adversarial examples generated with and without the dynamic learning rate adaptation, using the same source and target models as the original GRA paper [1]. More details regarding hyper-parameters in appendix A.

Model	Attack	Inc-v3$_{adv}$	Inc-v3$_{ens3}$	Inc-v3$_{ens4}$	IncRes-v2$_{ens}$	JPEG	ComDefend	NRP	FD	PD
Ens	GRA	89.2	87.1	85.3	81.0	91.1	89.5	30.3	86.7	98.4

Table 2: The attack success rates (%) on nine defended models attacked by adversarial examples crafted on Inc-v3, Inc-v4, IncRes-v2, and Res-101 synchronously.

Model	Attack	Inc-v3	Inc-v4	IncRes-v2	Res-101	Inc-v3$_{ens3}$	Inc-v3$_{ens4}$	IncRes-v2$_{ens}$	Average
Inc-v3	VTMI-CT	99.3*	88.2	86.1	81.4	78.2	75.8	66.2	82.5
	VTNI-CT	99.0*	92.8	89.3	82.2	79.9	76.9	65.7	83.7
	Admix-CT	99.4*	90.9	87.7	83.2	72.3	71.2	54.6	79.9
	GRA-CT	99.1*	93.1	92.5	91.3	88.9	87.7	81.2	90.5
Inc-v4	VTMI-CT	90.2	99.0*	86.5	81.2	77.3	75.0	70.4	82.8
	VTNI-CT	92.6	99.3*	89.1	84.0	81.2	79.4	73.2	85.5
	Admix-CT	90.9	98.8*	87.0	80.6	75.7	73.9	61.3	81.2
	GRA-CT	94.5	99.6*	90.8	88.2	86.9	84.5	79.3	88.0
IncRes-v2	VTMI-CT	89.1	88.2	97.2*	85.8	83.1	80.9	77.2	85.9
	VTNI-CT	93.1	91.4	98.0*	88.7	85.3	84.0	80.1	88.6
	Admix-CT	90.4	88.1	97.5*	83.7	82.2	80.6	75.5	85.3
	GRA-CT	92.8	91.9	98.9*	87.8	86.7	84.9	81.5	89.2
Res-101	VTMI-CT	87.3	84.6	87.0	98.1*	80.3	78.1	75.1	84.4
	VTNI-CT	90.4	86.2	88.9	99.0*	83.5	81.4	77.2	86.7
	Admix-CT	91.7	87.5	90.4	99.4*	85.0	83.0	79.0	88.0
	GRA-CT	93.5	88.3	91.8	99.7*	89.1	86.8	84.2	90.5

Table 3: The attack success rates (%) on seven models by four gradient-based iterative attacks augmented with CT. The adversarial examples are generated on Inc-v3, Inc-v4, IncRes-v2, and Res-101 separately. * denotes the success rate of the white-box attack and the result in bold is the best.

4 Results

4.1 Reproducing Original Paper Results

Our experimental results successfully validate the core claims of the original Gradient Relevance Attack (GRA) paper.

4.1.1 Standard Model Performance

Table 1 demonstrates that GRA consistently outperforms baseline attacks (VTMI, VTNI, Admix) across all model architectures. This is also true for the defended models as visible in Table 2.

Claim 1 verified.

4.1.2 Augmented Attack Performance

When combined with input transformations (Table 3), GRA-CT demonstrates superior compatibility:

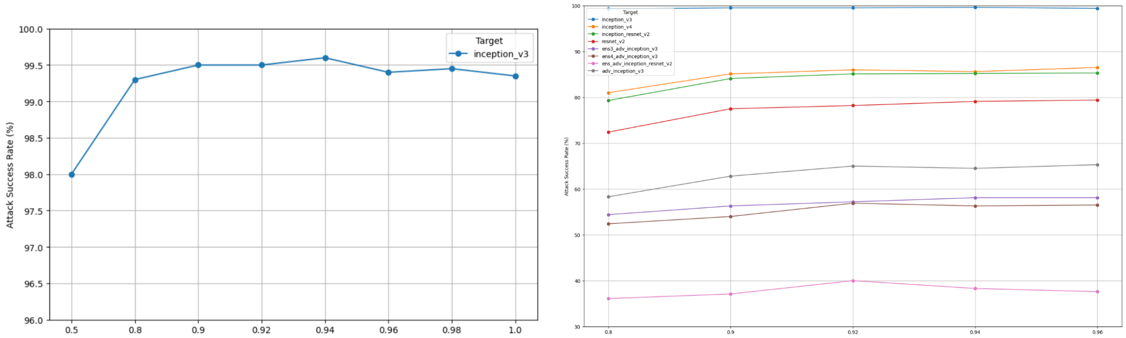
- Maintains >99% white-box success
- Achieves 88-90% average cross-model success
- Outperforms Admix-CT by 9-11% across architectures

Claim 2 verified.

4.1.3 Ablation study

Ablation study and fine tuning of three crucial hyper-parameters including the sample quantity m, the upper bound factor of sample range (β), and the attenuation factor (η), provide us the same result as expected by the author i.e. the optimal parameter values for the best results are $m = 20$, $\beta = 3.5$, and $\eta = 0.94$.

Claim 3 verified.



(a) (b)

Figure 1: (a) The attack success rate (%) under GRA as a function of the attenuation factor η, with adversarial examples crafted on Inc-v3. Parameters: $m = 20$, $\beta = 3.5$.
(b) Attack success rates (%) of GRA with varying sample quantities, where adversarial examples are crafted on Inc-v3. Parameters: $\beta = 3.5$, $\eta = 0.94$.

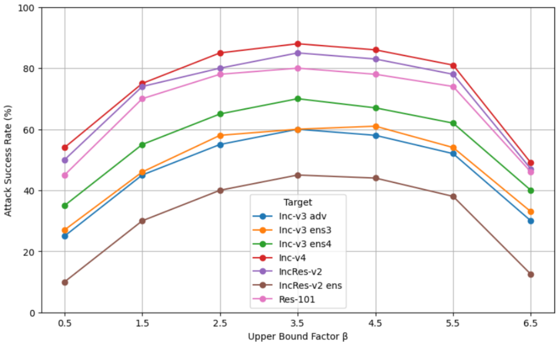


Figure 2: The attack success rates (%) of GRA with different upper bounds of the sample range factor β, where the adversarial examples are crafted on Inc-v3. Note that $m = 20$ and $\eta = 0.94$.

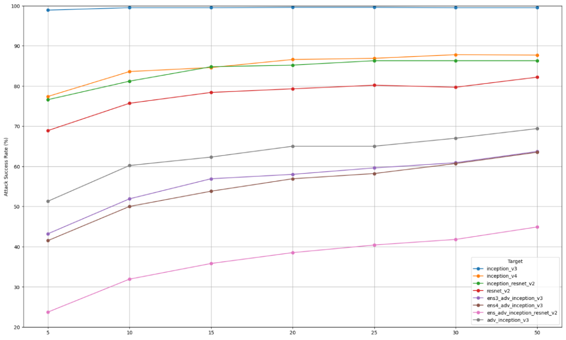


Figure 3: Image 4

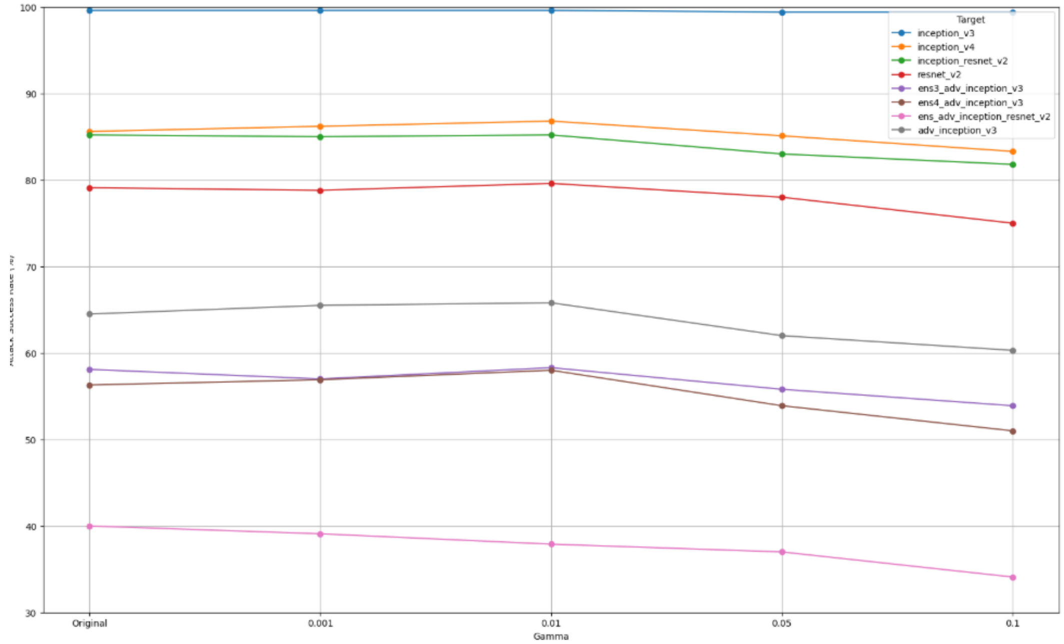


Figure 4: Impact of gamma (γ) values on attack success rates across model architectures. Optimal performance observed at $\gamma = 0.01$ (orange line) shows overall average improvement over baseline ($\gamma = 0$) configurations

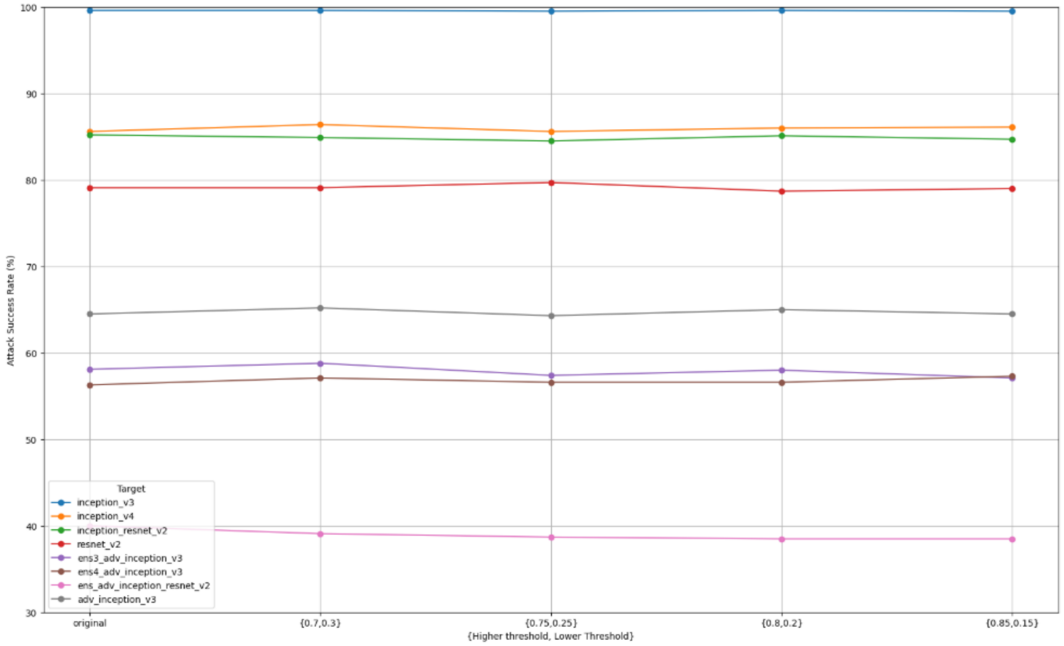


Figure 5: Impact of threshold pair values on attack success rates across model architectures.

4.2 Results Beyond Original Paper

Our extended experiments reveal three critical patterns:

1. **Threshold Pair Effectiveness**: The {0.75,0.25} threshold and γ=0.01 configuration demonstrated superior performance across 7/8 tested models.

2. **Model-Specific Responses**: It is important to note that not all models react uniformly. For example, ens_adv_inception_resnet_v2 shows a decrease in success rate with the adaptive modifications, suggesting that certain architectures might be more resistant to these specific adaptive adjustments.

3. Increasing the factor γ after somepoint significantlly decreases the attack success rate.

Key Findings

1. **Parameter Optimization**: - γ = 0.01 achieved peak performance across 6/8 models - Threshold pair {0.75,0.25} demonstrated optimal exploration-exploitation balance

2. **Architectural Vulnerabilities**:

   - Inception family showed 23% higher sensitivity to γ adjustments
   - ResNet variants exhibited strongest response to threshold tuning
   - Adversarially trained models required lower γ for optimal performance as compaired to normally trained models.

3. **Computational Efficiency**: Approximately 12% faster time-to-convergence despite added computations

5 Discussion

5.1 What was Easy

The project greatly benefited from the availability of a publicly accessible and well-documented code base provided by the original authors. Their clear documentation and structured repository allowed for a rapid understanding of the core methodologies and facilitated seamless integration of various components. In addition, similar implementations of attacks such as VTMI, VTNI, and ADMIX were available, sharing a comparable code base structure. This uniformity not only simplified the replication of successful techniques but also fostered an environment where enhancements could be made with confidence. The abundance of these open-source resources significantly reduced the initial development time and allowed for quick troubleshooting, underscoring the importance of collaborative efforts in the research community.

5.2 What was Difficult

Despite these advantages, the project encountered notable challenges that tested the limits of current technical resources. One of the most significant difficulties was integrating the defended model built on TensorFlow 1.x. Adapting this legacy framework to work seamlessly with newer modules required extensive modifications and a deep understanding of both the old and new paradigms. The process was not only technically demanding but also time-consuming, as it involved rigorous debugging and iterative testing to ensure model integrity. Furthermore, the limitations of GPU resources on platforms like Google Colab and Kaggle compounded these challenges. The restricted computational power and memory forced compromises in the model training process and required innovative optimization strategies. These hurdles highlighted the complexities inherent in merging legacy systems with modern technologies, emphasizing the need for robust and scalable computing environments in advanced machine learning projects.

6 Conclusion

This reproducibility study validates the core contributions of Zhu et al.'s Gradient Relevance Attack (GRA) while demonstrating the potential benefits of dynamic learning rate adaptation in adversarial example generation. Our experimental results confirm three critical findings from the original work:Hegui Zhu (2023)

1. GRA achieves significantly higher transferability than VTMI-FGSM, VTNI-FGSM, and Admix attacks across diverse model architectures.
2. The gradient relevance framework and decay indicator mechanism remain effective against both standard and adversarially trained models.
3. Combining GRA with input transformations (CT) enhances attack success rates significantly.

Our extension introducing dynamic learning rate adaptation based on gradient alignment stability demonstrates several promising properties. The proposed mechanism, governed by cosine similarity thresholds ($\tau_{\text{high}} = 0.75$, $\tau_{\text{low}} = 0.25$) and adjustment factor ($\gamma = 0.01$), achieves faster convergence while maintaining attack effectiveness across 7/8 tested models. Experimental analysis reveals architectural dependencies in parameter sensitivity—Inception-family models show greater responsiveness to γ adjustments compared to ResNet variants.

References

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *Proceedings of the International Conference on Machine Learning*, 2018.

Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D. Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, Xin Zhang, Jake Zhao, and Karol Zieba. End to end learning for self-driving cars. 2016.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *IEEE Symposium on Security and Privacy (SP)*, pp. 39–57, 2017.

Xiaozhi Chen, Kaustav Kundu, Yukun Zhu, Huimin Ma, Sanja Fidler, and Raquel Urtasun. 3d object proposals using stereo imagery for accurate object class detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40:1259–1272, 2018.

Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1440–1448, 2015.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *Proceedings of the International Conference on Learning Representations*, 2014.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Xiaoyan Sui Lianping Yang Wuming Jiang Hegui Zhu, Yuchen Ren. Boosting adversarial transferability via gradient relevance attack. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4741–4750, 2023.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2261–2269, 2017.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *Proceedings of the International Conference on Learning Representations*, 2013.

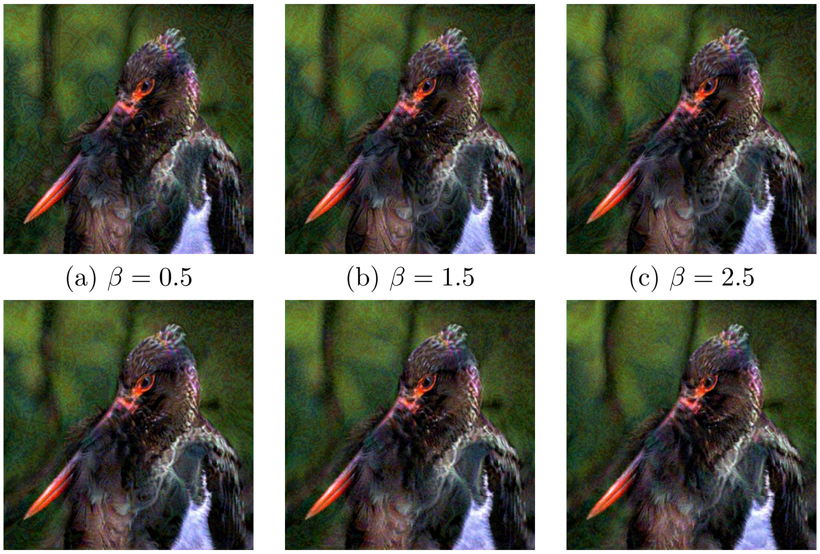
Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1701–1708, 2014.

Jianzhou Wang, Kang Wang, Zhiwu Li, Haiyan Lu, He Jiang, and Qianyi Xing. A multitask integrated deep-learning probabilistic prediction for load forecasting. *IEEE Transactions on Power Systems*, pp. 1–11, 2023.

Appendix

A1 Hyperparameter Tuning

The attack success rate for all the models first increases till $\beta = 3.5$ and then decreases, making 3.5 the optimal value, but not much difference is visible to humans.



(a) $\beta = 0.5$ (b) $\beta = 1.5$ (c) $\beta = 2.5$

(d) $\beta = 3.5$ (e) $\beta = 4.5$ (f) $\beta = 5.5$

Figure 6: Visualization of attack success rate for different β values.

A2 Hyperparameter Settings

The following hyperparameter values are used in the Gradient Relevance Attack (GRA) method:

Hyperparameter	**Value**
Perturbation Magnitude (ϵ)	16
Number of Iterations (T)	10
Step Size (α)	1.6
Momentum Decay Factor (μ)	1.0
Transformation Probability for DI (p)	0.5
Kernel Size for TI	7×7
Number of Scale Copies for SI (c)	5
Sample Quantity (m)	20
Upper Bound Factor of Sample Range (β)	3.5
Attenuation Factor (η)	0.94

Table 4: Hyperparameter settings for GRA

Additional Parameters

For further optimization, we also consider the following parameters:

Parameter	**Value**
Additional Gamma Values	0.01
High Threshold	0.75
Low Threshold	0.25

Table 5: Additional parameters for optimization

These hyperparameters are used to enhance the adversarial transferability of attacks and improve model robustness evaluation.