Paper Review :
Visual Prompting for Adversarial Robustness

Ravialdy Hidayat

Brief Recap of Adversarial Attacks

Generate adversarial examples (attacked images) :

Adversarial examples

Adversarial attack
(ex: small perturbation, *invisible by human eyes)*

\+

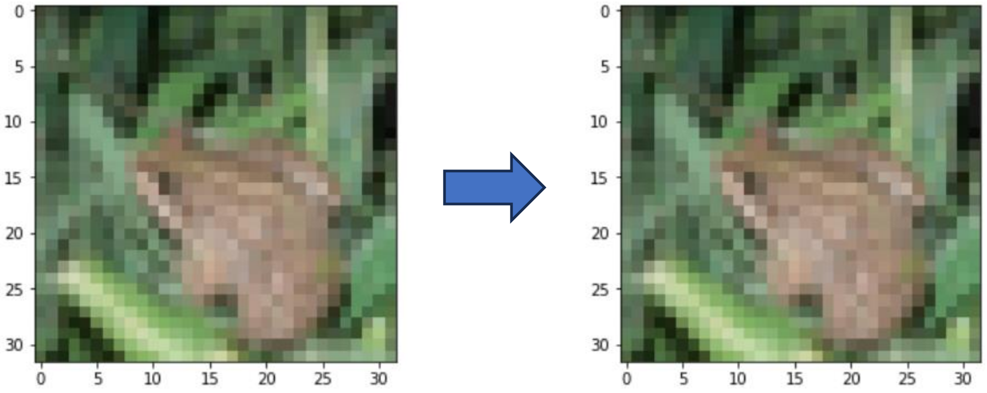
Clean image

→ Image recognition model → Significant accuracy drop

Main objective :

- **Fool** the **model** with an image **similar** to the original image.

Original Image | Original Image + Adversarial Attack



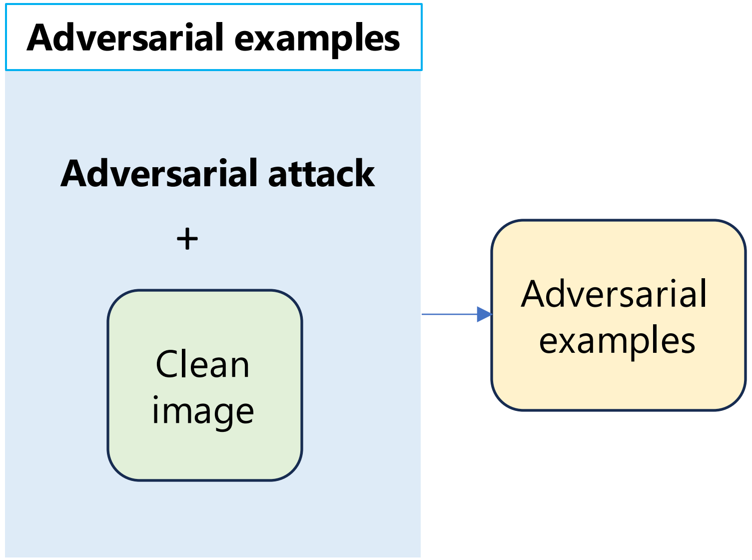
Prediction : Frog | Prediction : Panda

Brief Recap of Adversarial Training

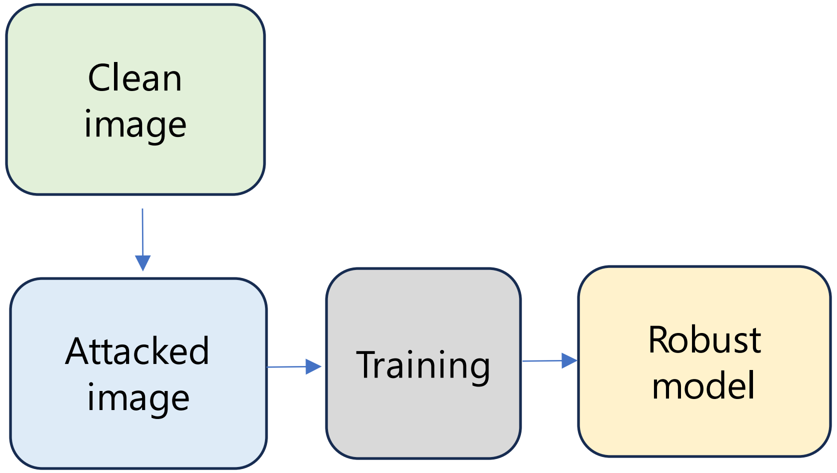
1. Model initialization.

Model architecture

2. Generate adversarial examples (attacked images).



3. Training the robust model.



4. Evaluation.
   - Test set on clean images.
   - Test set on attacked images.

Brief Recap of Test-Time Defense

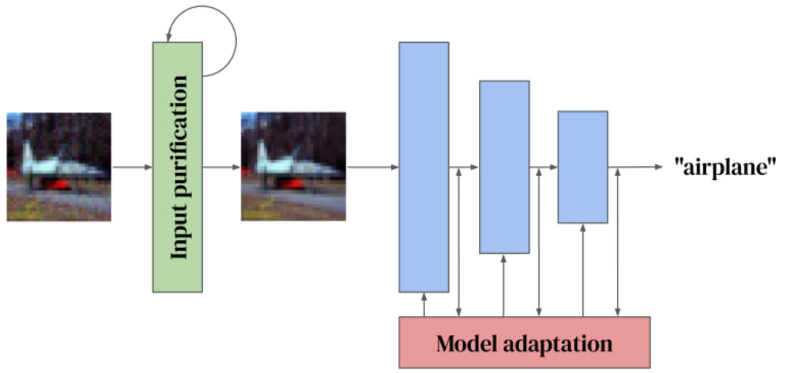


Figure 1. Different test-time defenses methods [23].

- Either purify the input via test-time augmentation or modify the model parameters [23].
- Input purification : Adding additional defense perturbation layer to the model (white-box or black-box) [24, 25]
- Model adaptation : Has access to the model parameters -> Only update some params while keeping most of it frozen.

Problems with Previous Works

- Adversarial Training : Needs to generate adversarial image for every/most input -> Massive computational cost [7, 8, 9, 10, 21, 23, 24].
- Test-Time Defense : Significantly increase the inference time [17, 23].

Brief Recap of Visual Prompting [26]

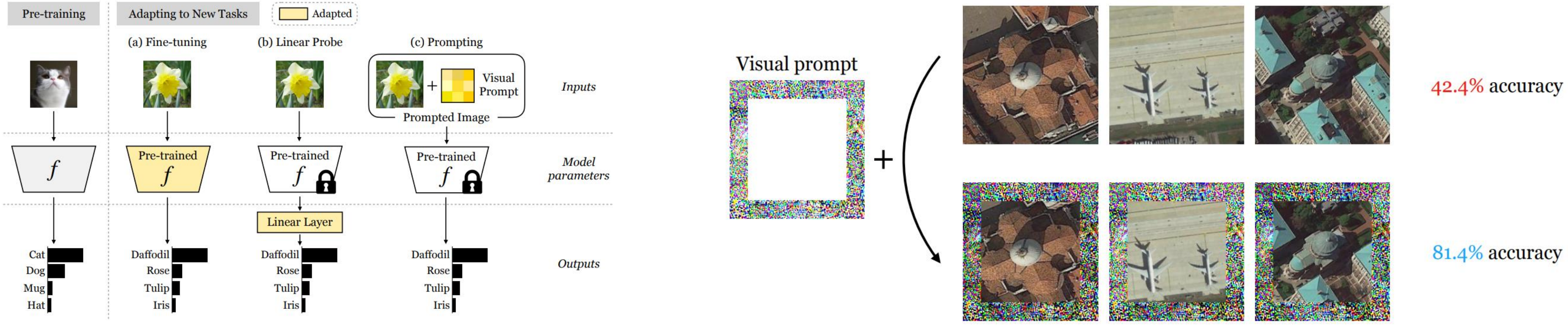


Figure 2. Illustration of visual prompting proposed by [26].

- Inspired by text prompting -> Leverage input space only to do transfer-learning.
- Successfully increased the performance on downstream task compared with zero-shot prediction.

Visual Prompting for Efficient Test-Time Defense [17] !

- Leverage Visual Prompting (VP) [26] to improve inference time for test-time defense.
- Achieve up to 42x inference time speed up compared to previous test-time defense methods [17].
- Originally defined as follows :

$$
\begin{array}{rl}
\text{Given:} & \mathcal{D}_{\text{tr}} \text{ as the training set.} \\
 & (\mathbf{x}, y) \text{ are feature } \mathbf{x} \text{ and label } y. \\
 & \ell \text{ as the error for training data.} \\
 & \boldsymbol{\theta} \text{ as the base model parameters.} \\
 & \mathcal{C} \text{ as the perturbation constraint set.} \\
\text{Find:} & \boldsymbol{\delta} \text{ as the visual prompt to be designed.} \\
\text{Objective:} & \underset{\boldsymbol{\delta}}{\text{minimize}} \quad \mathbb{E}_{(\mathbf{x},y)\in\mathcal{D}_{\text{tr}}}[\ell(\mathbf{x}+\boldsymbol{\delta}; y, \boldsymbol{\theta})] \\
\text{Subject to:} & \boldsymbol{\delta} \in \mathcal{C}
\end{array}
$$

Figure 3. Original optimization problem of vanilla VP [26].

Not A Straightforward Approach [17]

- Extend the concept of VP for adversarial robustness.
- Straightforward approach : Combine adversarial loss with generalization loss.

$$\begin{aligned}
\text{Given:}\quad & \mathcal{D}_{\mathrm{tr}} \text{ as the training set.} \\
& \epsilon \text{ as the radius for the } \ell_\infty\text{-norm ball.} \\
& \ell \text{ as the prediction error for training data.} \\
\text{Find:}\quad & \mathbf{x}' \text{ as the adversarial input.} \\
\text{Objective:}\quad & \ell_{\mathrm{adv}}(\mathbf{x}+\boldsymbol{\delta}; y, \boldsymbol{\theta}) = \text{maximize}_{\mathbf{x}':\|\mathbf{x}'-\mathbf{x}\|_\infty \le \epsilon}\, \ell(\mathbf{x}'+\boldsymbol{\delta}; y, \boldsymbol{\theta}) \\
\text{Subject to:}\quad & \mathbf{x}' \in \mathcal{B}_\epsilon(\mathbf{x}), \text{ where } \mathcal{B}_\epsilon(\mathbf{x}) \text{ is the } \ell_\infty\text{-norm ball at } \mathbf{x}.
\end{aligned}$$

$$\begin{aligned}
\text{Given:}\quad & \mathcal{D}_{\mathrm{tr}} \text{ as the training set.} \\
& \lambda \text{ as the regularization parameter.} \\
\text{Find:}\quad & \boldsymbol{\delta} \text{ as the visual prompt to be designed.} \\
\text{Objective:}\quad & \underset{\boldsymbol{\delta}}{\text{minimize}} \quad \lambda \mathbb{E}_{(\mathbf{x},y)\in\mathcal{D}_{\mathrm{tr}}}[\ell(\mathbf{x}+\boldsymbol{\delta}; y, \boldsymbol{\theta})] + \\
& \mathbb{E}_{(\mathbf{x},y)\in\mathcal{D}_{\mathrm{tr}}}[\ell_{\mathrm{adv}}(\mathbf{x}+\boldsymbol{\delta}; y, \boldsymbol{\theta})] \\
\text{Subject to:}\quad & \boldsymbol{\delta} \in \mathcal{C}
\end{aligned}$$

Figure 4. Optimization problem of U-AVP [17].

- *Note : Regularization parameter to balance between generalization and adversarial robustness.
- Called Universal AVP (U-AVP). Can be solved with common min-max optimization method.

Problems with Universal Adversarial Visual Prompt (U-AVP) [17]

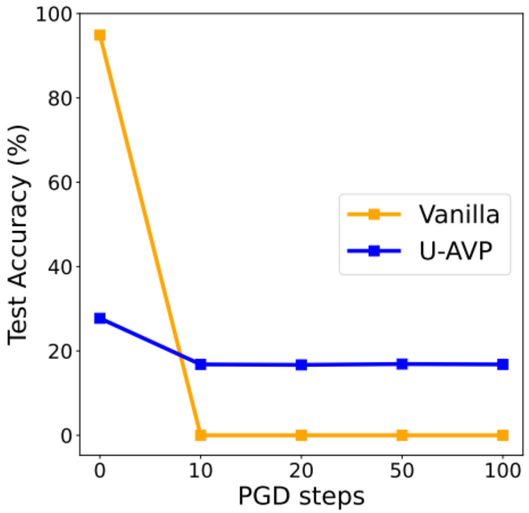


Figure 5. Performance of U-AVP compared with vanilla VP [17].

- Dropped significantly in terms of standard accuracy (PGD step = 0).
- Not quite robust in terms of robustness accuracy (only improve ~18%).
- Reason : Due to same visual prompt for all inputs.

Problems with Direct Extension of U-AVP (C-AVP-v0) [17]

$$\begin{aligned}
&\text{Given:} && \mathcal{D}_{\mathrm{tr}} \text{ split into } \left\{\mathcal{D}_{\mathrm{tr}}^{(i)}\right\}_{i=1}^{N} \text{ for } N \text{ classes.} \\
& && \ell_{\mathrm{adv}} \text{ as the adversarial error for training data.} \\
&\text{Find:} && \left\{\boldsymbol{\delta}^{(i)}\right\}_{i\in[N]} \text{ as the class-wise visual prompts.} \\
&\text{Objective:} && \underset{\left\{\boldsymbol{\delta}^{(i)}\in\mathcal{C}\right\}_{i\in[N]}}{\text{minimize}} \quad \frac{1}{N}\sum_{i=1}^{N}\left\{\lambda\mathbb{E}_{(\mathbf{x},y)\in\mathcal{D}_{\mathrm{tr}}^{(i)}}\left[\ell\left(\mathbf{x}+\boldsymbol{\delta}^{(i)}; y, \boldsymbol{\theta}\right)\right] + \mathbb{E}_{(\mathbf{x},y)\in\mathcal{D}_{\mathrm{tr}}^{(i)}}\left[\ell_{\mathrm{adv}}\left(\mathbf{x}+\boldsymbol{\delta}^{(i)}; y, \boldsymbol{\theta}\right)\right]\right\}
\end{aligned}$$

Figure 6. Optimization problem of C-AVP-v0 [17].

- Leverages model's prediction to choose class-specific visual prompt.
- Lead to very poor prediction accuracy.
- Can serve as backdoor attack trigger [26] if the model's prediction is incorrect.
- Called C-AVP-v0 (Class-wise Adversarial Visual Prompt zeroth version).

Proposed Idea : Joint Optimization for C-AVP ! [17]

$$\ell_{\mathrm{C-AVP},1}(\{\boldsymbol{\delta}^{(i)}\};\mathcal{D}_{\mathrm{tr}},\boldsymbol{\theta}) = \mathbb{E}_{(\mathbf{x},y)\in\mathcal{D}_{\mathrm{tr}}} \max\{\max_{k\neq y} f_k(\mathbf{x}+\boldsymbol{\delta}^{(k)};\boldsymbol{\theta}) - f_y(\mathbf{x}+\boldsymbol{\delta}^{(y)};\boldsymbol{\theta}), -\tau\},$$

$$\ell_{\mathrm{C-AVP},2}(\{\boldsymbol{\delta}^{(i)}\};\mathcal{D}_{\mathrm{tr}},\boldsymbol{\theta}) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{(\mathbf{x},y)\in\mathcal{D}_{\mathrm{tr}}^{(-i)}} \max\{f_i(\mathbf{x}+\boldsymbol{\delta}^{(i)};\boldsymbol{\theta}) - f_y(\mathbf{x}+\boldsymbol{\delta}^{(i)};\boldsymbol{\theta}), -\tau\},$$

$$\ell_{\mathrm{C-AVP},3}(\{\boldsymbol{\delta}^{(i)}\};\mathcal{D}_{\mathrm{tr}},\boldsymbol{\theta}) = \mathbb{E}_{(\mathbf{x},y)\in\mathcal{D}_{\mathrm{tr}}} \max\{\max_{k\neq y} f_y(\mathbf{x}+\boldsymbol{\delta}^{(k)};\boldsymbol{\theta}) - f_y(\mathbf{x}+\boldsymbol{\delta}^{(y)};\boldsymbol{\theta}), -\tau\}.$$

Given: $\mathcal{D}_{\mathrm{tr}}$ split into $\left\{\mathcal{D}_{\mathrm{tr}}^{(i)}\right\}_{i=1}^{N}$ for N classes.

τ as the confidence threshold.

γ as a parameter for class-wise prompting penalties.

Find: $\left\{\boldsymbol{\delta}^{(i)}\right\}_{i\in[N]}$ as the class-wise visual prompts.

Objective:
$$\underset{\left\{\boldsymbol{\delta}^{(i)}\in\mathcal{C}\right\}_{i\in[N]}}{\text{minimize}} \quad \ell_{\mathrm{C-AVP},0}\left(\left\{\boldsymbol{\delta}^{(i)}\right\};\mathcal{D}_{\mathrm{tr}},\boldsymbol{\theta}\right) + \gamma\sum_{q=1}^{3}\ell_{\mathrm{C-AVP},q}\left(\left\{\boldsymbol{\delta}^{(i)}\right\};\mathcal{D}_{\mathrm{tr}},\boldsymbol{\theta}\right)$$

N : Total number of classes,

i : Index for a specific class in $[N]$,

k : Class not equal to y,

y : True class label

Figure 7. Joint optimization problem proposed by [17].

- Introduce 3 additional losses to avoid backdoor attack trigger phenomenon.
- Simultaneously optimize class-specific visual prompts to not only enhance correct classifications but also minimize backdoor-like behaviors.

Performance and Limitations [17]

Evaluation metrics (%)	Std acc	Robust acc vs PGD w/ step # 10	20	50	100
Pre-trained	**94.92**	0	0	0	0
Vanilla VP	94.48	0	0	0	0
U-AVP	27.75	16.9	16.81	16.81	16.7
C-AVP-v0	19.69	13.91	13.63	13.6	13.58
C-AVP (ours)	57.57	**34.75**	**34.62**	**34.51**	**33.63**

Figure 8. Table performance stated by [17].

- Significantly improve robustness accuracy compared with vanilla VP.
- Still lag behind from vanilla VP in terms of standard accuracy.
- Only tested on CIFAR-10 dataset.

References :

[1] Liu et al., “SignSGD via Zeroth-Order Oracle”, International Conference on Learning Representations (ICLR), 2019

[2] Guo et al., "Simple Black-box Adversarial Attacks", International Conference on Machine Learning (ICML), 2019

[3] Uesato et al., "Adversarial Risk and the Dangers of Evaluating Against Weak Attacks", International Conference on Machine Learning (ICML), 2018

[4] Ilyas et al., “Prior Convictions: Black-Box Adversarial Attacks with Bandits and Priors”, International Conference on Learning Representations (ICLR), 2019

[5] Madry et al., “Towards Deep Learning Models Resistant to Adversarial Attacks”, International Conference on Learning Representations (ICLR), 2018

[6] Spall et al., “A Stochastic Approximation Technique for Generating Maximum Likelihood Parameter Estimates”, Proceedings of the American Control Conference, 1987

[7] Zheng et al., “Efficient Adversarial Training with Transferable Adversarial Examples”, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020

[8] Hadi et al., “ℓ_∞-Robustness and Beyond: Unleashing Efficient Adversarial Training”, European Conference on Computer Vision (ECCV), 2020

References :

[9] Wu et al., "Towards Efficient Adversarial Training on Vision Transformers", European Conference on Computer Vision (ECCV), 2020

[10] Xi et al., "Efficient Adversarial Training with Robust Early-Bird Tickets", Conference on Empirical Methods in Natural Language Processing (EMNLP), 2022

[11] Zhang et al., "How to Robustify Black-Box ML Models? A Zeroth-Order Optimization Perspective", International Conference on Learning Representations (ICLR), 2022

[12] Yoon et al., "Adversarial purification with Score-based generative models", International Conference on Machine Learning (ICML), 2018

[13] Shi et al., "Online Adversarial Purification based on Self-Supervision", International Conference on Learning Representations (ICLR), 2021

[14] Chen et al., "Towards Robust Neural Networks via Close-loop Control", International Conference on Learning Representations (ICLR), 2021

[15] Carlini et al., "Adversarial Examples Are Not Easily Detected: Bypassing Ten Detection Methods", Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security, 2017

[16] Oh et al., "BlackVIP: Black-Box Visual Prompting for Robust Transfer Learning", EEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020

References :

[17] Chen et al., "Visual Prompting for Adversarial Robustness.", International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2023

[18] Salman et al., "Denoised Smoothing: A Provable Defense for Pretrained Classifiers.", Conference on Neural Information Processing Systems (NeurIPS), 2020

[19] Kumar et al., "Model Inversion Networks for Model-Based Optimization.", Conference on Neural Information Processing Systems (NeurIPS), 2020

[20] Oh et al., "Towards reverse-engineering black-box neural networks.", In Explainable AI: Interpreting, Explaining and Visualizing Deep Learning, pp. 121–144. Springer, 2019

[21] Zhang et al., "The Limitations of Adversarial Training and the Blind-Spot Attack.", International Conference on Learning Representations (ICLR), 2019

[22] Carlini et al., "(CERTIFIED!!) ADVERSARIAL ROBUSTNESS FOR FREE!", International Conference on Learning Representations (ICLR), 2023

[23] Croce et al., "Evaluating the Adversarial Robustness of Adaptive Test-time Defenses.", International Conference on Machine Learning (ICML), 2022

References :

[24] Alfarra et al., “Combating adversaries with antiadversaries.”, AAAI, 2022

[25] Wang et al., “Dynamic defenses against adversarial attacks.”, arXiv:2105.08714, 2021

[25] Bahng et al., “Visual prompting: Modifying pixel space to adapt pre-trained models.”, arXiv:2203.17274, 2022.

[26] Gu et al., “Badnets: Identifying vulnerabilities in the machine learning model supply chain.”, arXiv:1708.06733, 2017.