GuardianDream: Advanced Defense Mechanisms Against Deepfake Exploitation in DreamBooth

MIPAL 2024 Winter Internship Project Final Presentation

Mijin Koo, Saehee Eom

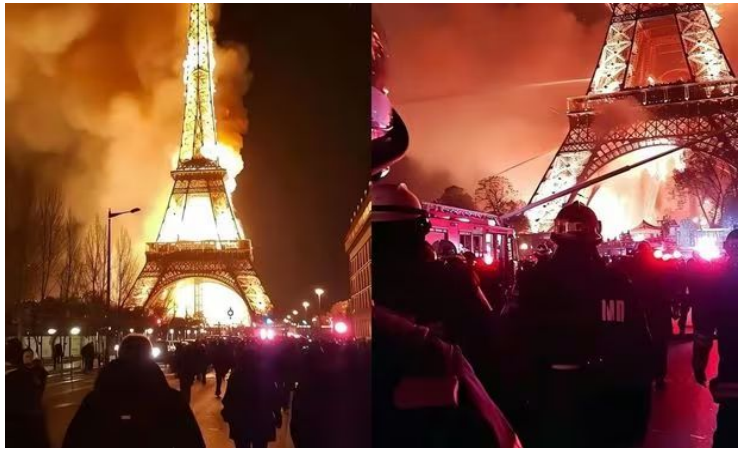
2024-02-29

Table of Contents

Introduction

Background

- Advancement of Generative Models: GAN, DIffusion
- High-Quality Personalized and Customized AIGC
- Dangers of DEEPFAKE (ex. Synthesis of Fake News and Explicit Content)



Fake photo created by AI

Introduction

Project Overview

“**GuardianDream**: Advanced Defense Mechanisms Against Deepfake Exploitation in DreamBooth”

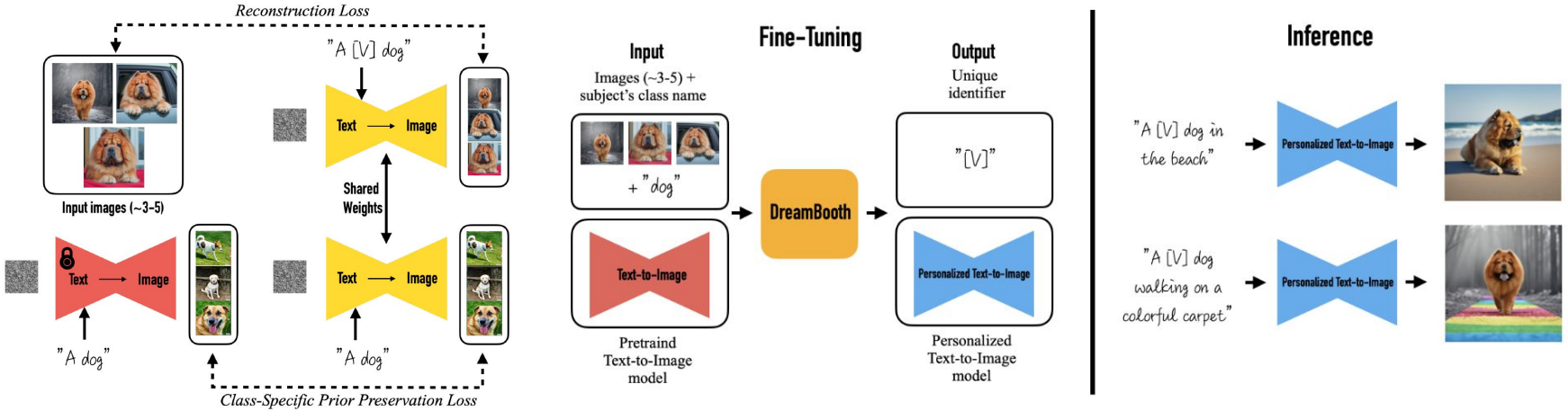
- Propose advanced defense mechanisms against Deepfake exploitation in DreamBooth[1], a Stable Diffusion based personalization model
- Our goal is to find adversarial noise that defeats personlization
- Aimed to validate and advance the defense method proposed by Anti-DreamBooth[2]

[1] DreamBooth: Fine Tuning Text-to-Image Diffusion Models for Subject-Driven Generation, CVPR, 2023
[2] Anti-DreamBooth: Protecting users from personalized text-to-image synthesis, ICCV, 2023

Prior Studies

Personalization of text-to-image diffusion models

- [Goal] To personalize text-to-image diffusion models for instance of interest
- Use rare-token identifier to make model learn to bind a specific subject
- Training loss combines two objectives, reconstruction loss and prior preservation loss



DreamBooth: Fine Tuning Text-to-Image Diffusion Models for Subject-Driven Generation, CVPR, 2023

Prior Studies

Adversarial Attack

- [Goal] To find an imperceptible perturbation of an input image to mislead the behavior of given models
- The minimal visual difference is enforced by $\| x' - x \|_p < \eta$, denoted by $\Delta = \{\delta : \| \delta \|_p < \eta\}$
- Find the optimal perturba

$$\delta_{\text{adv}} = \arg\max_{\delta \in \Delta} \mathcal{L}(f(x+\delta), y_{\text{true}})$$

ss in the untargeted version:

- FGSM[1] computes gradients of the model's loss with respect to the input image. PGD[2] method iterative

$$x'_0 = x$$
$$x'_k = \Pi_{(x,\eta)}(x'_{k-1} + \alpha \cdot \text{sgn}(\nabla_x \mathcal{L}(f(x'_{k-1}), y_{true})))$$

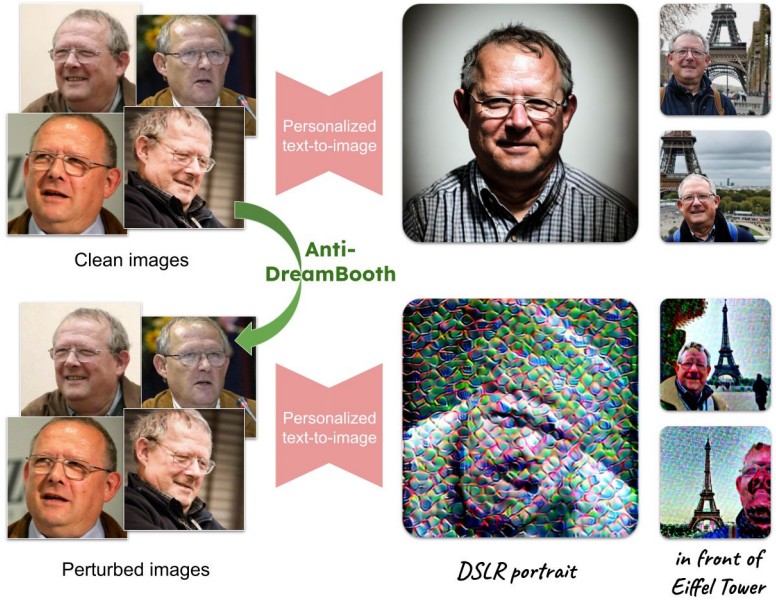
incrementally.

[1] Explaining and Harnessing Adversarial Examples, ICLR, 2015
[2] Towards Deep Learning Models Resistant to Adversarial Attacks, ICLR, 2018

Prior Studies

Anti-DreamBooth

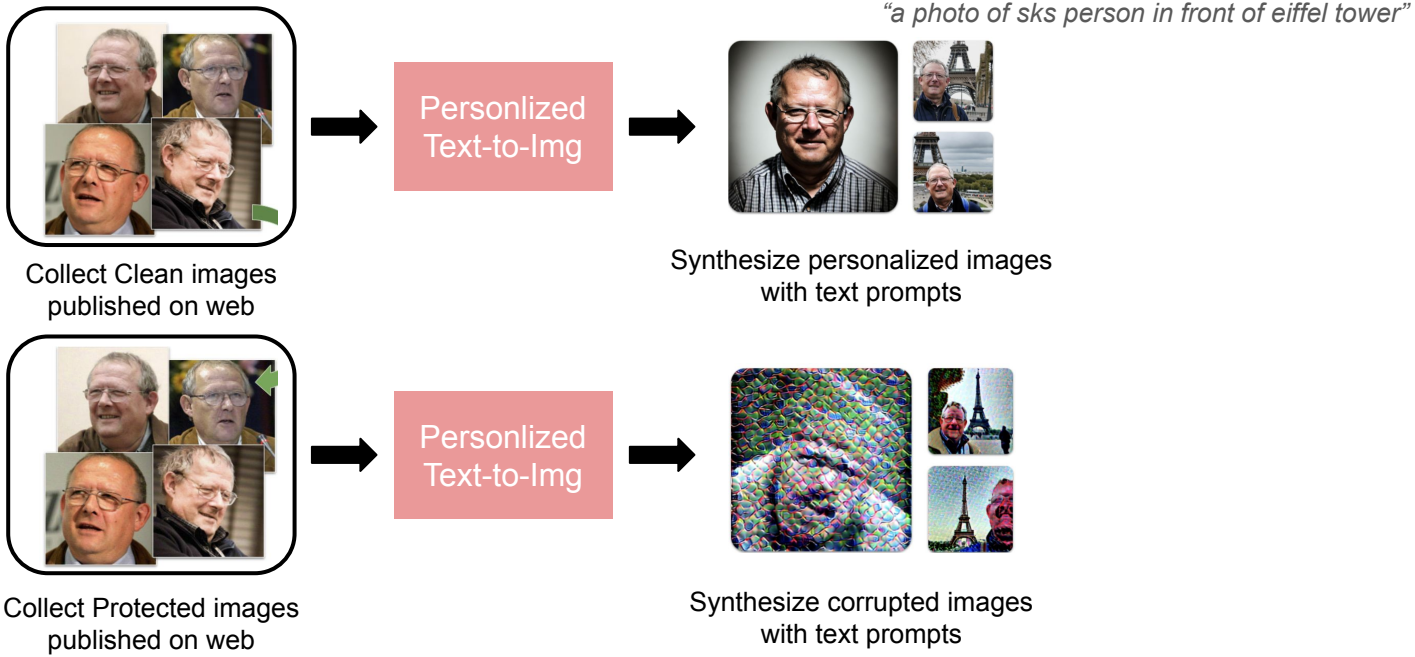
- [Goal] Craft imperceptible perturbations for each user's image, disrupting DreamBooth models
- Proposed defense mechanism, called FSMG(Fully-trained Surrogate Model Guidance), to find adversarial noise



Prior Studies

Anti-DreamBooth

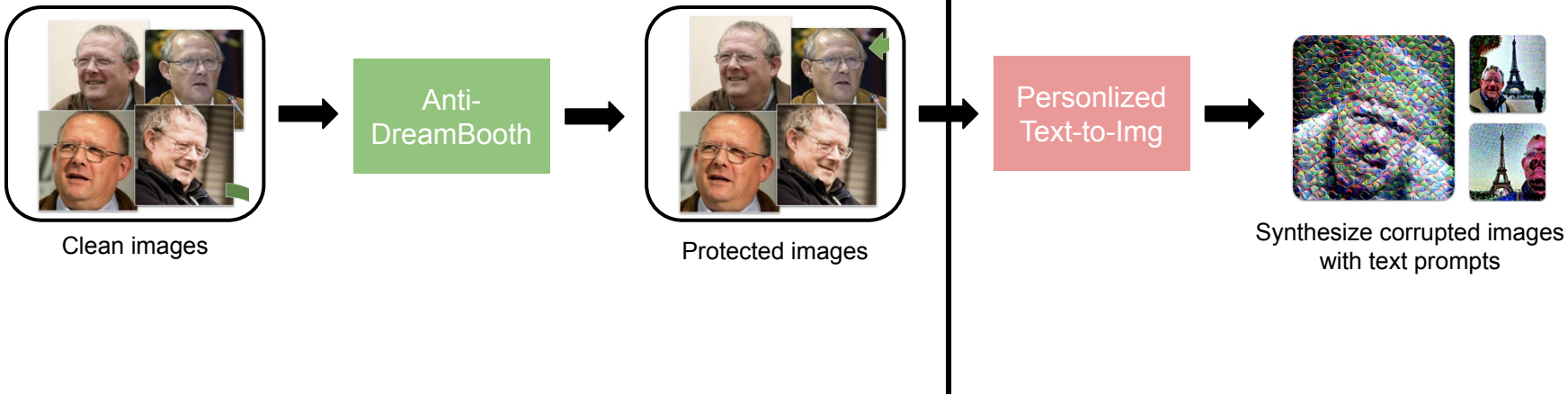
- Deepfake Scenario (Attacker's view)



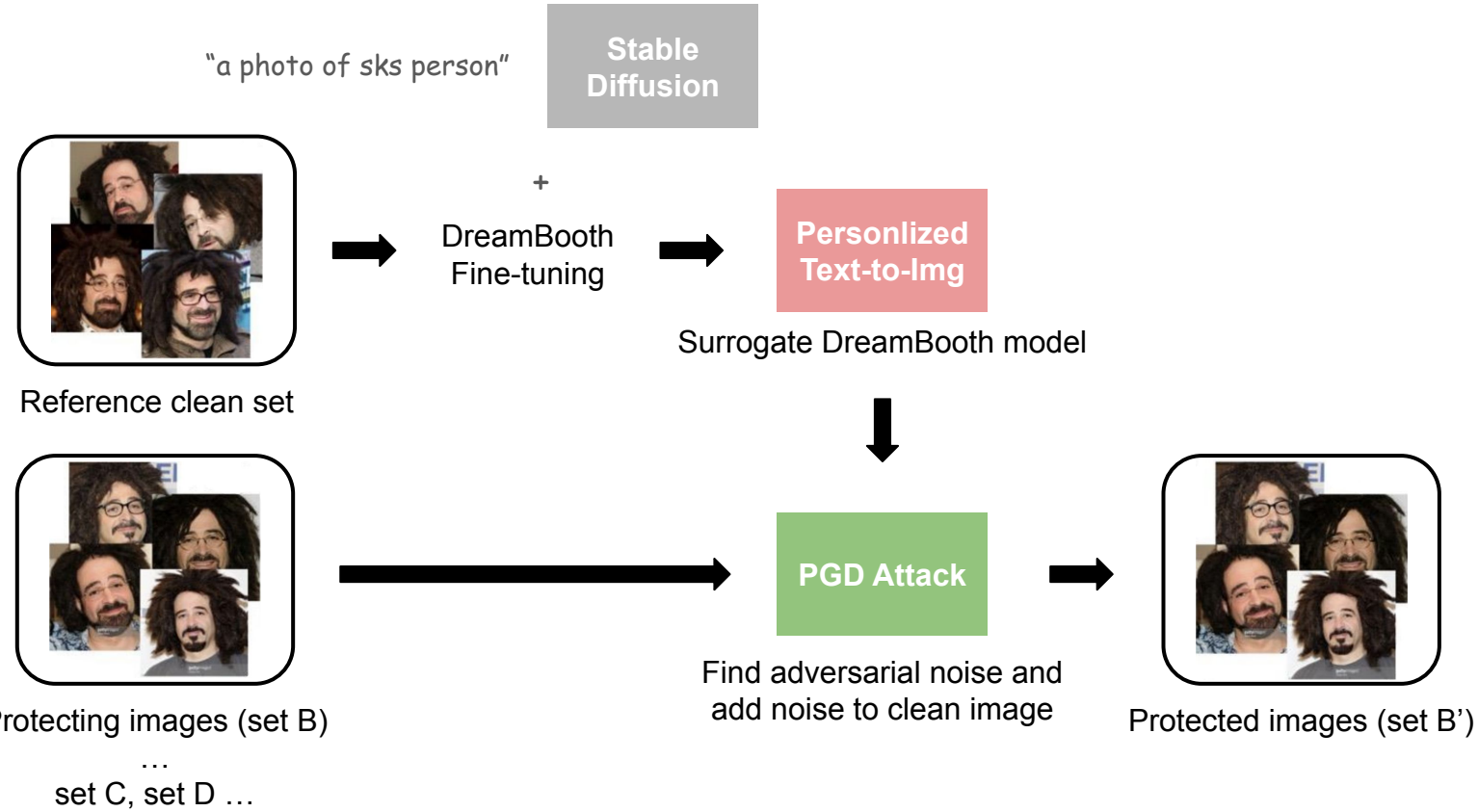
Prior Studies

Anti-DreamBooth

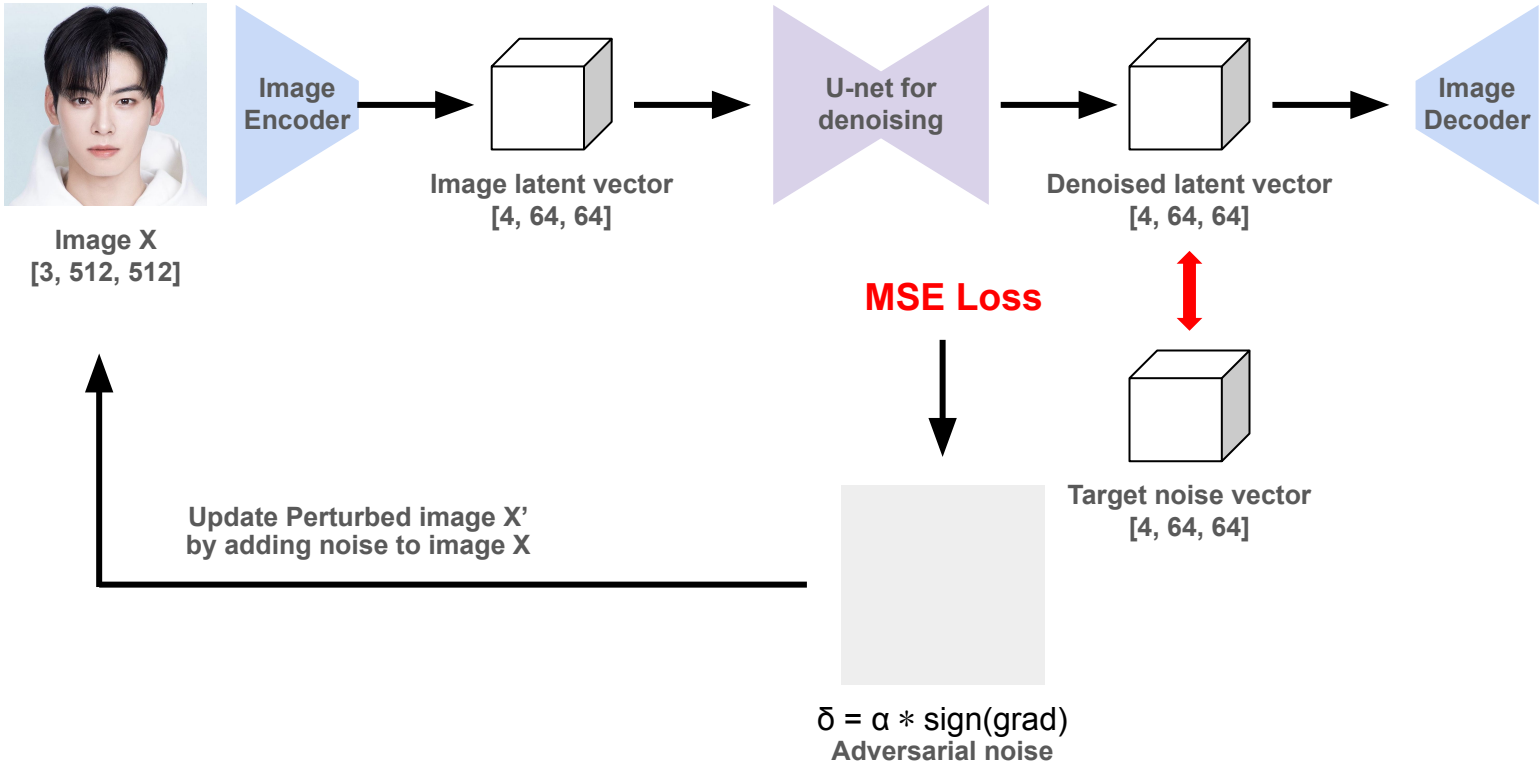
- Defense Scenario (Defender's view)



Prior Studies



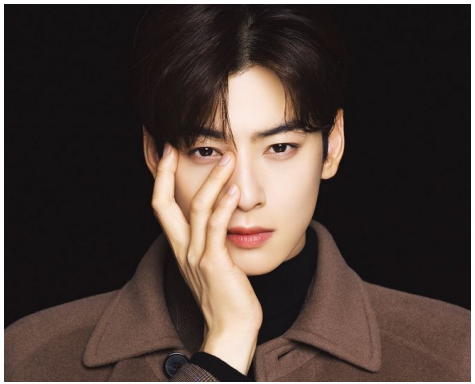
Prior Studies



Prior Method Validation

DreamBooth

By testing various learning rates, training steps, and inference steps, we explored the optimal training and inference parameters possible within the available computing resources.




▲ Images used to train the DreamBooth model

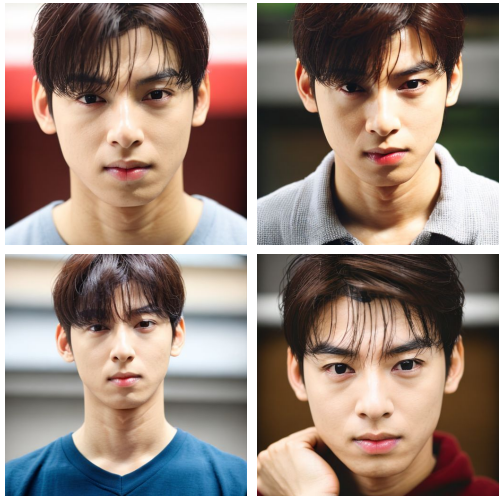
Prior Method Validation

DreamBooth

By testing various learning rates, training steps, and inference steps, we explored the optimal training and inference parameters possible within the available computing resources.

▲lr: 5e-7, inference 100 steps

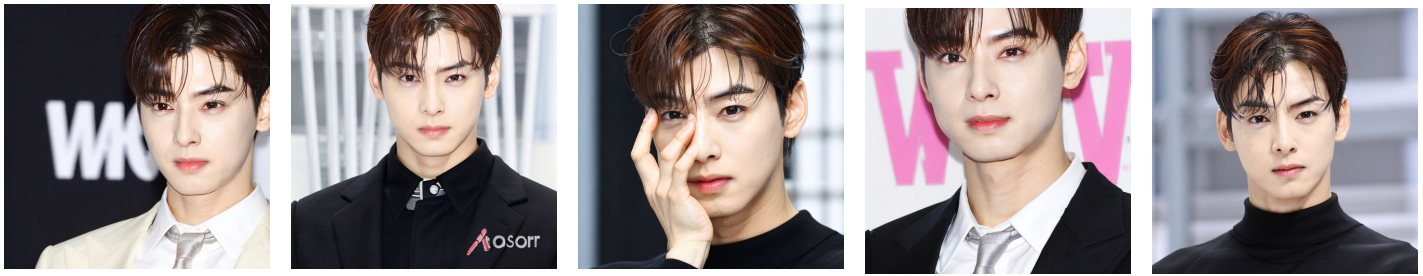
▲lr: 5e-7, inference 200 steps



▲lr: 1e-6, inference 200 steps

Prior Method Validation

DreamBooth



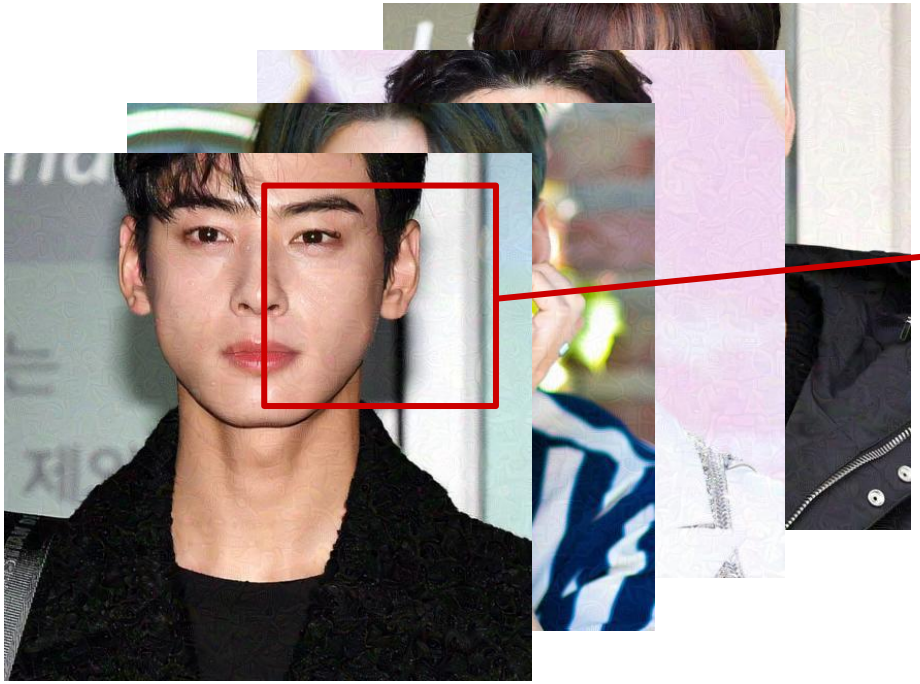
"a photo of sks person"

"a photo of sks person
with kim jong-un in north korea"

"a photo of sks person in jail"

Prior Method Validation

Vanilla FSMG (noise budget: 0.05)

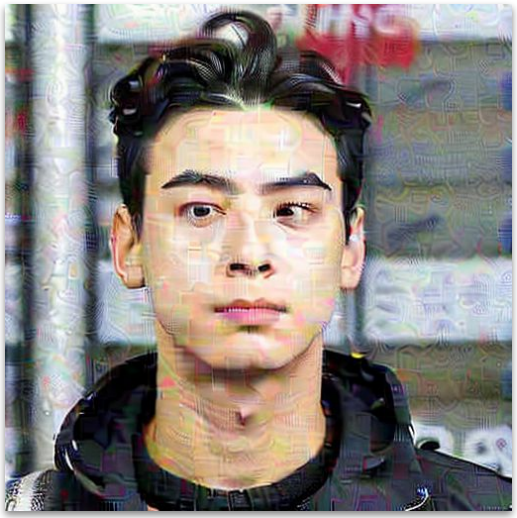


Prior Method Validation

Vanilla FSMG (noise budget: 0.05)

"a photo of sks person"

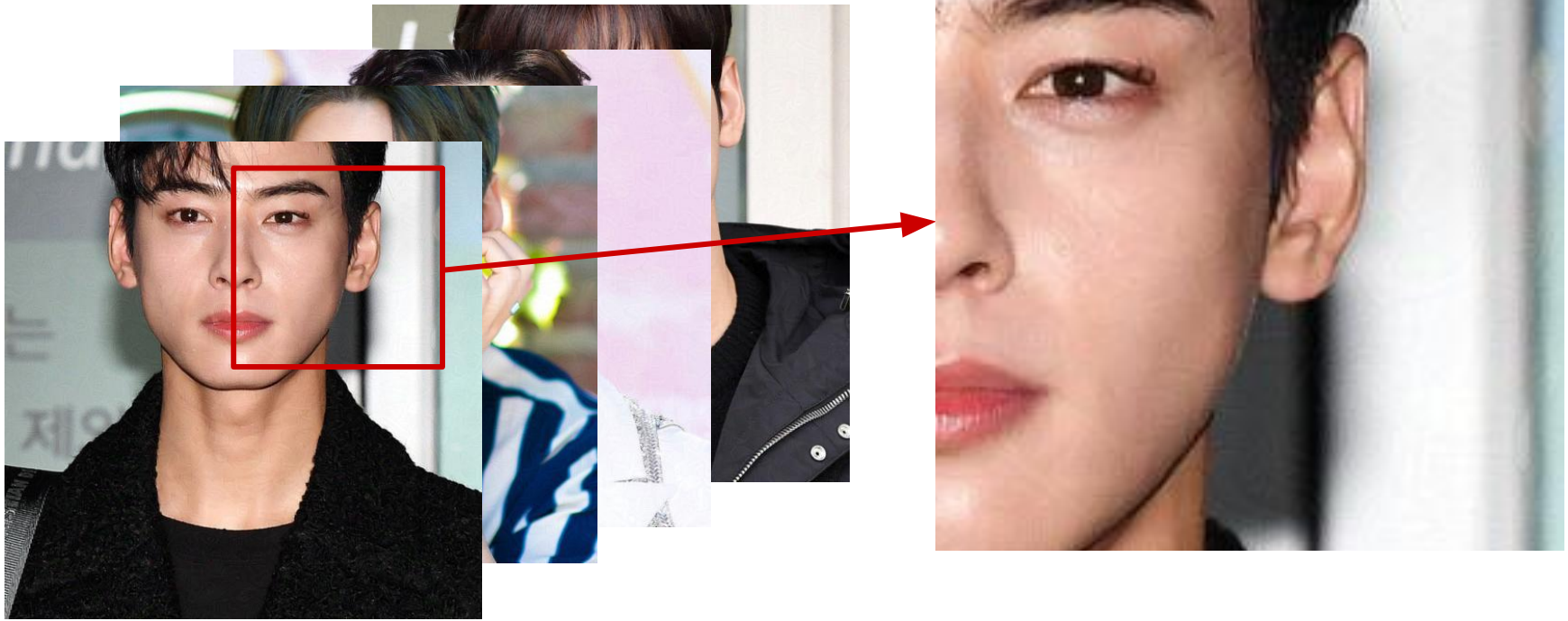
"a photo of sks person
with kim jong-un in north korea"



"a photo of sks person in jail"

Prior Method Validation

Vanilla FSMG (noise budget: 0.02)



Prior Method Validation

Vanilla FSMG (noise budget: 0.02)

"a photo of sks person"

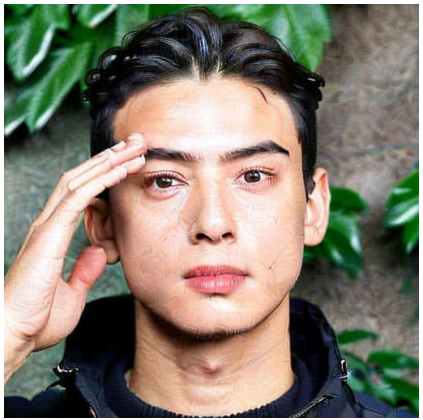
"a photo of sks person
with kim jong-un in north korea"

"a photo of sks person in jail"

Prior Method Validation

Vanilla FSMG

- [Goal] Low image quality / unrecognizable face / prompt inapplicable
- [Limitations]
  - Defense fails in some cases
  - Noise is visible to human eye

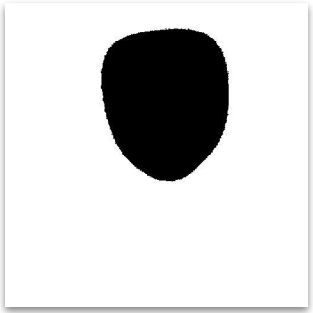


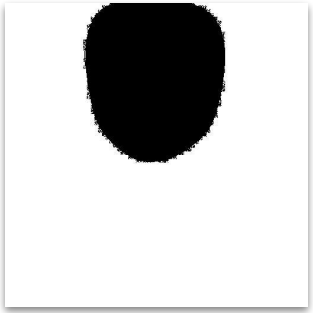
"a photo of sks person in the jungle"
noise budget: 0.02

fsmg training image
noise budget: 0.05

Improvement Idea 1: face mask noise control

- **Image quality - defense tradeoff:** The resulting images from Anti-DreamBooth show that the larger the noise size, the better the protection, but the more visible the noise is to the human eye.
- We aim to find a way to circumvent this tradeoff, maintaining a level of noise that provides sufficient defense while keeping the noise imperceptible.
- This involves identifying the face of the person being protected by Anti-DreamBooth **and applying less noise, or no noise at all, to the face area.**
- Example images (face detection: using MediaPipe face landmark detection)



Improvement Idea 1: face mask noise control

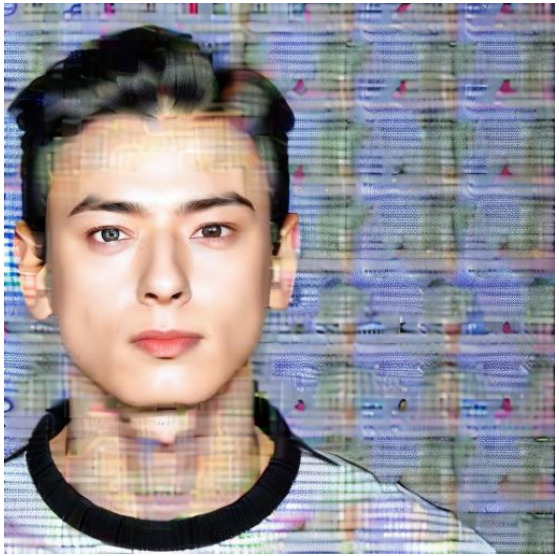
Experiments

1. Noise budget **in** face: 0, **out**: 0.05

2. Noise budget **in** face: 0.02, **out**: 0.05

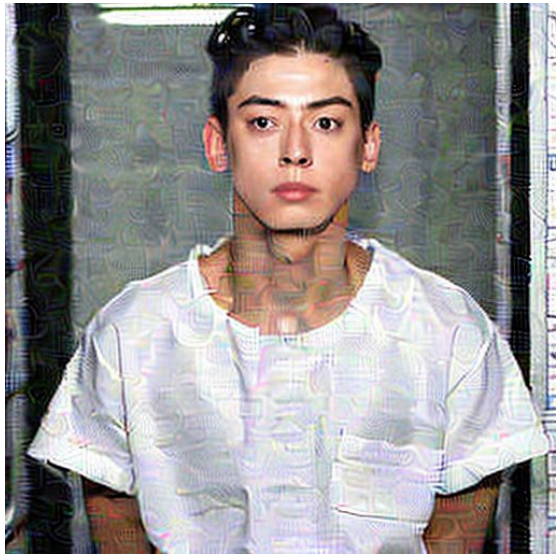
Improvement Idea 1: face mask noise control

Experiments - 1. Noise budget **in** face: 0, **out**: 0.05



"a dslr portrait of sks person"

"a photo of sks person with kim jong-un in north korea"



"a photo of sks person in jail"

Improvement Idea 1: face mask noise control

Experiments - 1. Noise budget **in** face: 0, **out**: 0.05

"a photo of sks person armed with big guns"

"a photo of sks person in mountain fuji"

"a photo of sks person
being arrested by the police"

"a photo of sks person
kissing a young lady"

Improvement Idea 1: face mask noise control

Experiments - 2. Noise budget **in** face: 0.02, **out**: 0.05

"a dslr portrait of sks person"

"a photo of sks person with kim jong-un in north korea"

"a photo of sks person in jail"

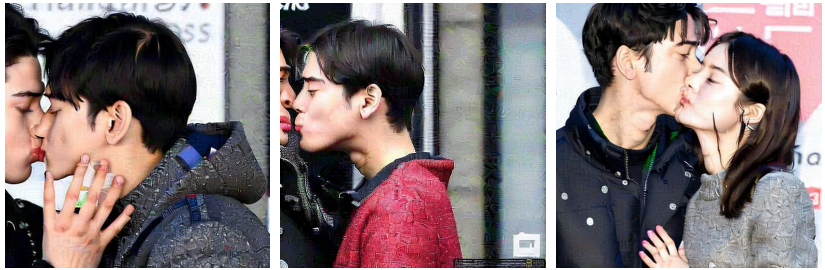
Improvement Idea 1: face mask noise control

Experiments - 2. Noise budget **in** face: 0.02, **out**: 0.05

"a photo of sks person armed with big guns"

"a photo of sks person in mountain fuji"

"a photo of sks person
being arrested by the police"



"a photo of sks person
kissing a young lady"

Improvement Idea 1: face mask noise control

Quantitative Results

- **Evaluation Metrics**
  - **Face Detection Failure Rate (FDFR):** Checked whether a face was detected using the RetinaFace detector.
  - **Identity Score Matching (ISM):** If a face was detected, the face recognition embedding was extracted using the ArcFace recognizer and compared to the average of the face embeddings of the faces in the clean image (cosine distance measured).
  - **SER-FIQ**: Facial image quality assessment metric.
- Measurements were taken on six inferred images for each of the 18 prompts for each model, and the average was calculated.
- A higher FDFR indicates a more successful defense method, while lower ISM and SER-FIQ values indicate better performance.

Improvement Idea 1: face mask noise control

Quantitative Results

	FDFR ↑	ISM ↓	SER-FIQ ↓
Vanilla FSMG 0.02	0.00	0.34	0.6278
Vanilla FSMG 0.05	**0.16**	**0.30**	0.6279
Face Mask 0	0.02	0.36	0.6267
Face Mask 0.02	0.05	**0.31**	**0.6213**

Vanilla FSMG 0.02 and *Vanilla FSMG 0.05* are existing Anti-DreamBooth FSMG models with noise budgets set to 0.02 and 0.05 respectively, while *Face Mask 0* and *Face Mask 0.02* are models trained with the noise budget for the face region set to 0 and 0.02 respectively.

Improvement Idea 2: Loss function for Face Similarity

- As a result of fine-tuning DreamBooth with distorted images generated by Anti-db, the subject identity was maintained while patterned noise was added to the images.
- The objective of the PGD attack was reset to fail at identity matching.
- The Deepface library was used to obtain face embedding vectors from the images, and a loss function was implemented using cosine similarity with the target embedding.

deepFace

Improvement Idea 2: Loss function for Face Similarity

Image X
[3, 512, 512]

Image Encoder

Image latent vector
[4, 64, 64]

U-net for denoising

Denoised latent vector
[4, 64, 64]

Image Decoder

Decoded Image
[3, 512, 512]

Face Encoder

Face embedding [512, 1]

Target embedding

MSE Loss

Cosine Similarity

$\delta = \alpha * \text{sign}(\text{grad})$
Adversarial noise

Update Perturbed image X'
by adding noise to image X

Conclusion

Results

- Developed methods to protect portrait photos from deepfakes
- Confirmed and improved the results of the methods proposed in previous research.

Future Research

- Develop universal noise that can be generally applied to portrait photos.
- Need to verify if it is robust against adversarial purification methods such as DiffPure[1].

[1] Diffusion Models for Adversarial Purification, ICML, 2022

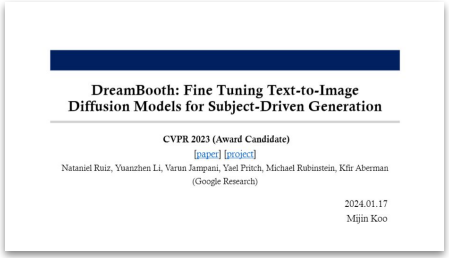
Summary of Internship Activities

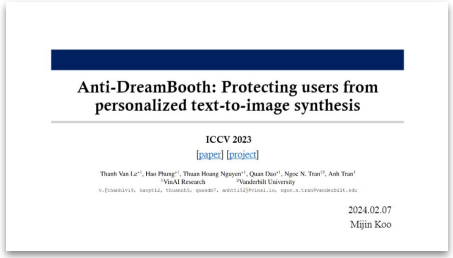
- **Project Timeline**

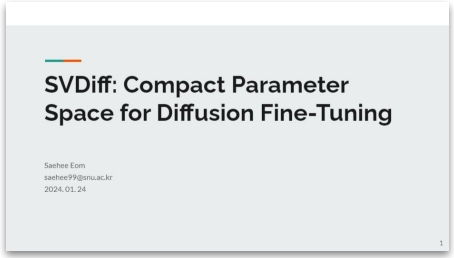
구분		Week								
업무	내용	1	2	3	4	5	6	7	8	9
프로젝트 주제 선정 및 팀 빌딩										
스터디										
선행 연구 리서치										
기존 Method 검증										
Sprint1	Model 개발 및 연구 Idea 검증									
	Fine Tuning									
	Validation									
Sprint2	Model 개발 및 연구 Idea 검증									
	Fine Tuning									
	Validation									
최종 평가	발표 준비									

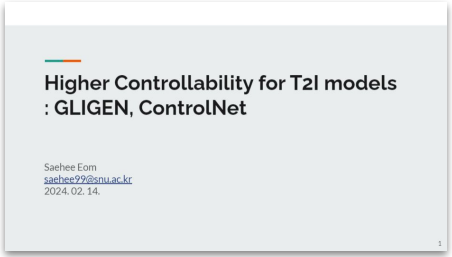
Summary of Internship Activities

- **Paper Review Sessions**
  - Mijin
    - DreamBooth: Fine Tuning Text-to-Image Diffusion Models for Subject-Driven Generation
    - Anti-DreamBooth: Protecting users from personalized text-to-image synthesis
  - Saehee
    - SVDiff: Compact Parameter Space for Diffusion Fine-Tuning
    - Higher Controllability for T2I models: GLIGEN, ControlNet









Thank You for Listening!

Appendix

Implementation of FSMG

```
for _ in range(pgd_train_steps):
  ...
  # Predict the noise residual
  model_pred = unet(noisy_latents, timesteps, encoder_hidden_states).sample

  # Get the target for loss depending on the prediction type
  target = noise

  # Calculate MSE loss between the predicted noise and the target noise
  loss = F.mse_loss(model_pred.float(), target.float(), reduction="mean")

  # Compute the gradient of the loss with respect to the perturbed images
  grad = torch.autograd.grad(loss, perturbed_images, retain_graph=False, create_graph=False)[0]

  # Add adversarial noise to image
  adv_images = pertubed_images + alpha * grad.sign()
```

Project Contributions

- Project topic selection, literature review research, existing method verification, presentation preparation: Mijin, Saehee
- Improvement Idea 1: Saehee
- Improvement Idea 2: Mijin