CS588 Final Project Presentation

Deepfake Retrieval Systems: Detecting Identity Fraud in Image Databases

Jumin Lee and Suhyeon Ha (T4)

2024. 06. 05.

SGVR Lab
KAIST

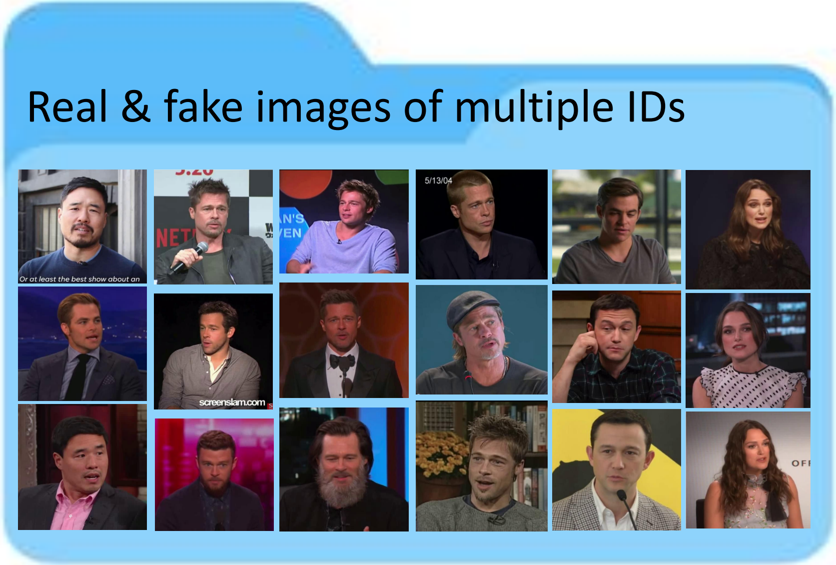
Introduction

Target Task

- Given an authentic image, our goal is to detect fake images pretending to depict the same person in database.

ID 13, Real

Query



Database

ID 13, Fake

Results

Our Framework

- No existing work to retrieve deepfakes of the query image.

- A combination of face retrieval and forgery detection can be utilized.

Stage 1	Stage 2
Face Retrieval	Forgery Detection

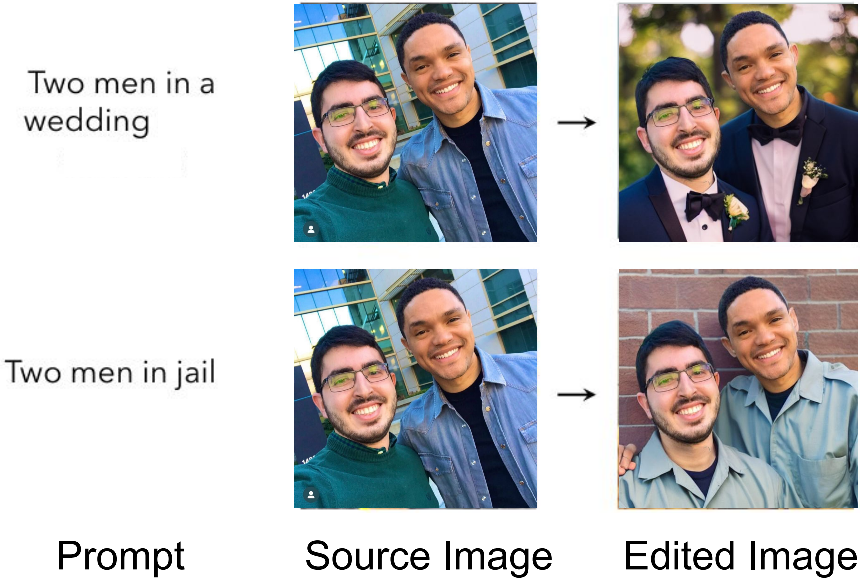
or

Stage 1	Stage 2
Forgery Detection	Face Retrieval

- Face retrieval
  : Identify images that match the given identity.
- Forgery detection
  : Determine whether the identified **arbitrary images** have been manipulated.

Our Framework

- Prompt-guided inpainting can modify images while preserving their identities.
- If we use <u>deepfake detection</u> instead of <u>forgery detection</u>, we can not handle this issue.



Our Framework

- No existing work to retrieve deepfakes of the query image.

- A combination of face retrieval and forgery detection can be utilized.

Stage 1	Stage 2
Face Retrieval	Forgery Detection

or

Stage 1	Stage 2
Forgery Detection	Face Retrieval

  - Why is face retrieval ahead of forgery detection?

    : Being unrecognized as someone's identity suggests its quality is doubtful.

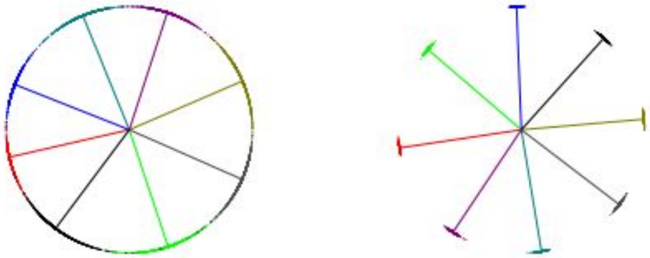
Our Framework

- Given an authentic image, our goal is to detect fake images pretending to depict the same person in database.

Related Work

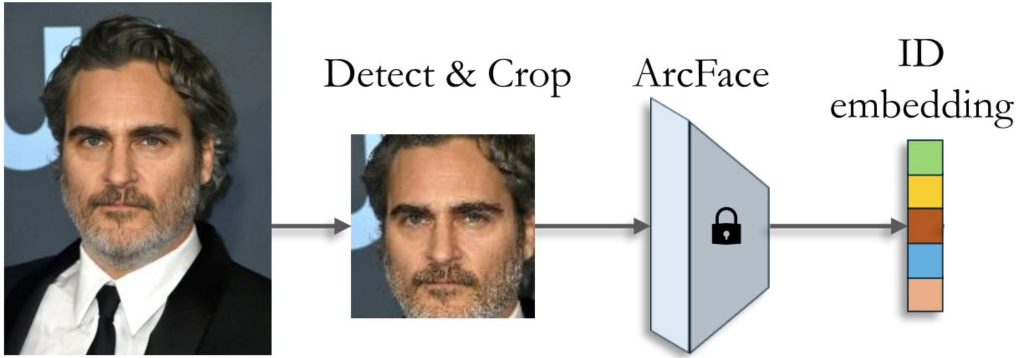
ArcFace

- ArcFace: Additive Angular Margin Loss for Deep Face Recognition, CVPR 2019



(a) Softmax (b) ArcFace

< Training >

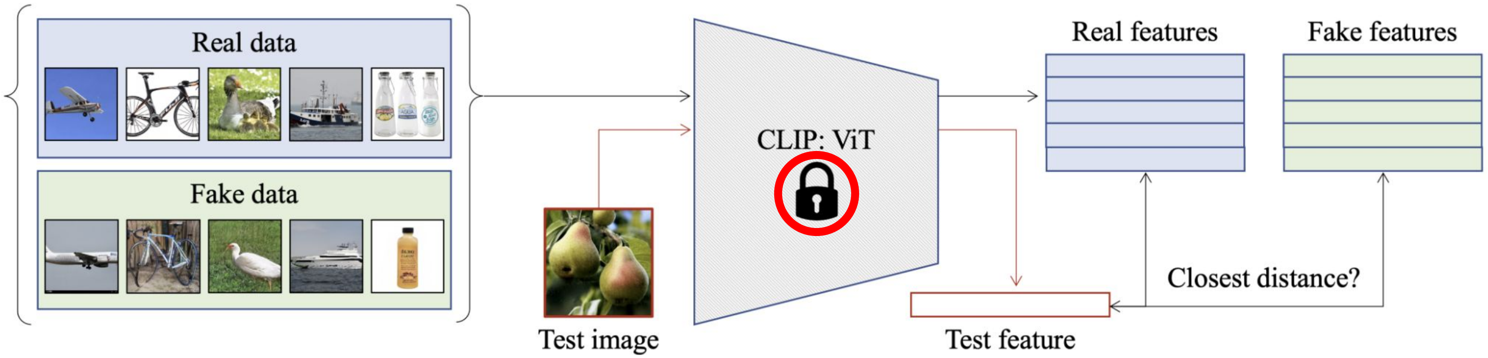


< Inference >

- Train with loss term that depends on the angle between classes to create a larger gap between different classes.
- After training, model can get ID embedding.

UniDet

- Towards Universal Fake Image Detectors that Generalize Across Generative Models, CVPR 23

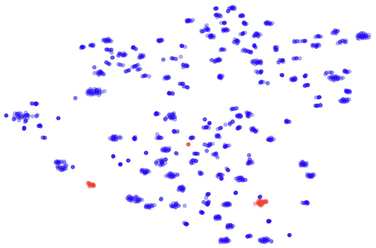
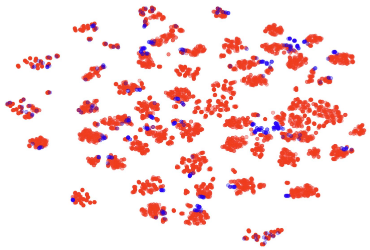


The classification process should happen in a **feature space** which <u>has not been trained</u> to separate images from the two classes.

In Midterm Project Presentation

- Checked the performance of UniDet for stage 2.
- Classification accuracy result.

Detection method	Generative Adversarial Networks						Denoising Diffusion Models			DALL-E	Deepfake Models	
	Pro-GAN	Cycle-GAN	Big-GAN	Style-GAN	Gau-GAN	Star-GAN	Glide	Guided	LDM		Deepfakes	CelebDF
UniDet w/ LC	100.00	**98.50**	**94.50**	82.00	**99.50**	97.00	79.07	70.03	**94.19**	81.47	66.60	11.01

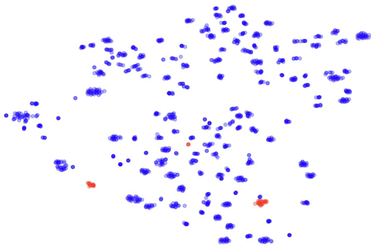
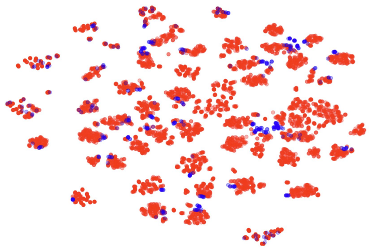
	Deepfakes	Celeb-DF
UniDet's image embedding (Real/Fake)		
Classification Accuracy	**66.60** (Real Accuracy = 94%, **Fake Accuracy = 42%**)	**11.01** (Real Accuracy = 99%, **Fake Accuracy = 1%**)

In Midterm Project Presentation

- There is still room for improvement in forgery detection for both datasets.

→ Our goal is to improve facial forgery detection of UniDet for deepfake retrieval system.

Detection method	Generative Adversarial Networks						Denoising Diffusion Models			DALL-E	Deepfake Models	
	Pro-GAN	Cycle-GAN	Big-GAN	Style-GAN	Gau-GAN	Star-GAN	Glide	Guided	LDM		Deepfakes	CelebDF
UniDet w/ LC	100.00	**98.50**	**94.50**	82.00	**99.50**	97.00	79.07	70.03	**94.19**	81.47	66.60	11.01

	Deepfakes	Celeb-DF
UniDet's image embedding (Real/Fake)		
Classification Accuracy	**66.60** (Real Accuracy = 94%, **Fake Accuracy = 42%**)	**11.01** (Real Accuracy = 99%, **Fake Accuracy = 1%**)

Our Approach

Our Approach

- UniDet uses only CLIP’s visual features for forgery detection.
- We try to combine CLIP's visual and **text features** to improve UniDet's performance.
    - It's used for many other tasks such as classification and generation.

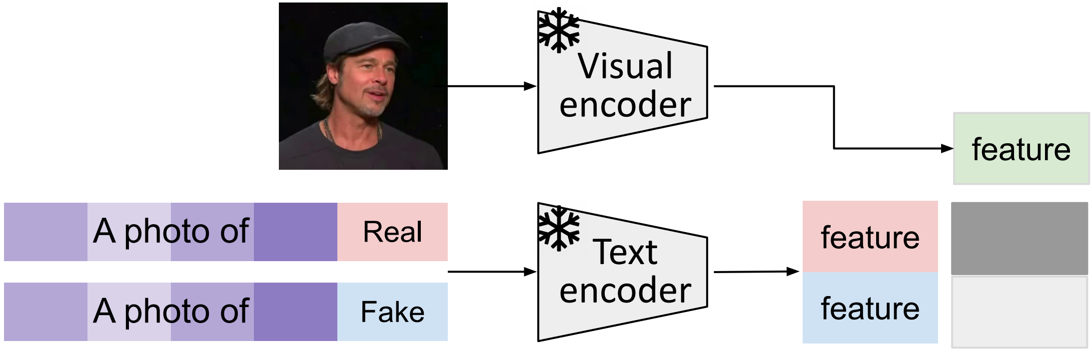
      ex) Classification : CoOp[1], CoCoOp[2]

      ex) Generation : Arc2Face[3]

[1] Learning to Prompt for Vision-Language Models, IJCV 2022
[2] Conditional Prompt Learning for Vision-Language Models, CVPR 2022
[3] Arc2Face: A Foundation Model of Human Faces, arxiv 2024

Our Approach

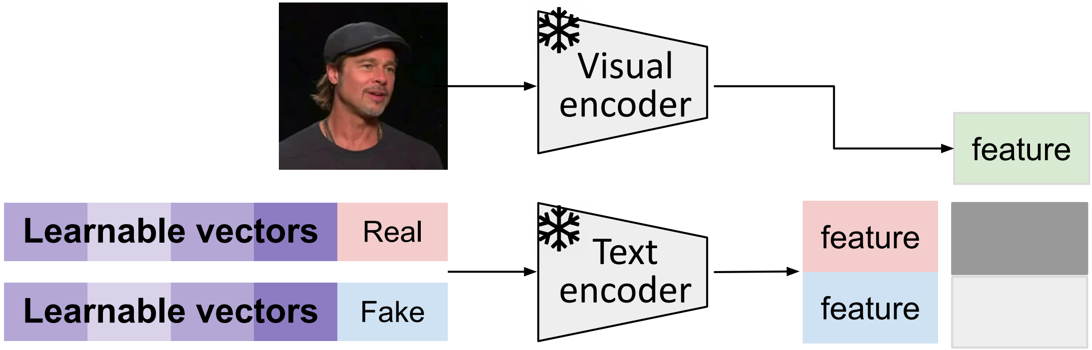
- UniDet uses only CLIP’s visual features for forgery detection.
- We try to combine CLIP's visual and **text features** to improve UniDet's performance.



- However, we need prompt engineering which is inefficient.
  - A photo of [CLASS]. / A photo of a [CLASS]. / A [CLASS]. / …

Our Approach

- UniDet uses only CLIP’s visual features for forgery detection.
- We try to combine CLIP's visual and **text features** to improve UniDet's performance.



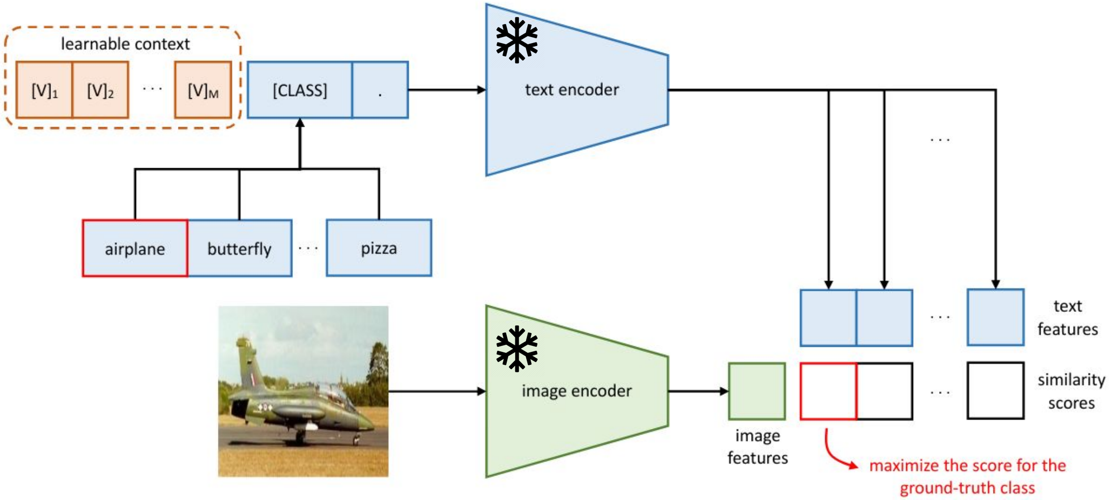
- So we apply Context Optimization (CoOp).

Our Approach

- What is Context Optimization (CoOp) ?

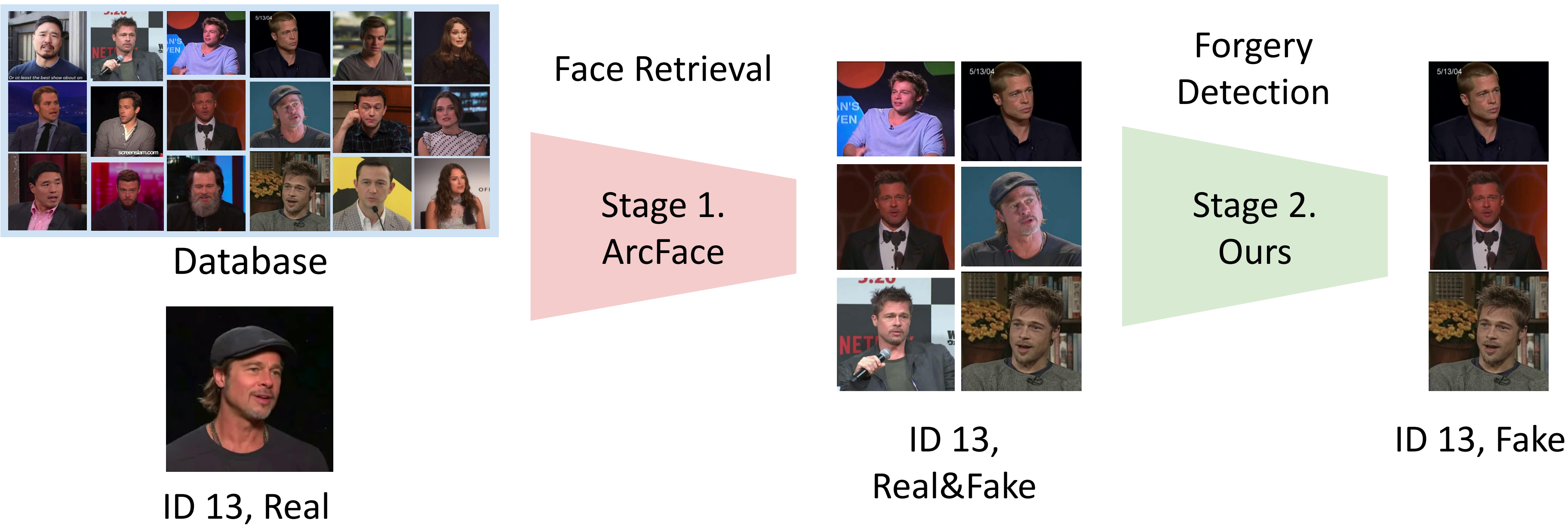
: Model prompt's context words with learnable vectors while the entire pre-trained parameters are kept fixed.

- Using cross entropy loss.



Our Framework

- Inference



Results

Experiment details for Stage 2.

- Training Dataset : ProGAN dataset
- Evaluation Dataset for Deepfake Models
  - Deepfakes : 5,405 frames (= 2,707 real + 2,698 fake)
  - CelebDF : 50,205 frames (= 4,820 real + 45,385 fake)

Real

Fake

Deepfakes

CelebDF

Quantitative results

- Stage 2. Classification Accuracy.

Detection method	Generative Adversarial Networks						Denoising Diffusion Models			DALL-E	Deepfake Models		Avg.
	Pro-GAN	Cycle-GAN	Big-GAN	Style-GAN	Gau-GAN	Star-GAN	Glide	Guided	LDM		Deepfakes	CelebDF	
UniDet w/ LC	100.00	**98.50**	**94.50**	82.00	**99.50**	97.00	79.07	70.03	**94.19**	81.47	66.60	11.01	81.15
Ours	100.00	95.60	93.80	**95.25**	93.43	**99.15**	**92.88**	**84.3**	88.16	**91.50**	**79.63**	**11.70**	**93.22**

  - Demonstrates **high performance gains**, especially on Deepfakes dataset.
  - However, we also have same problem with CelebDF dataset.

- Even if we crop out just the faces like in the deepfake dataset, we achieve **14.25%** accuracy performance.

CelebDF : 50,205 frames (= 4,820 real + 45,385 fake)

Quantitative results

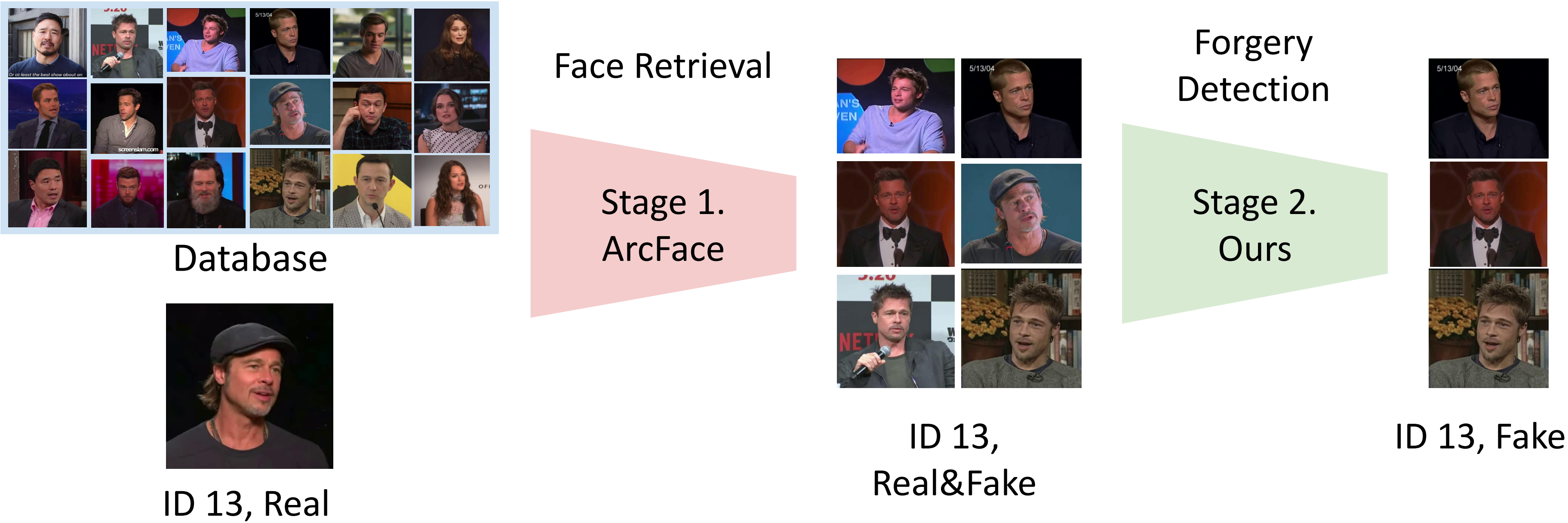
- Stage 2. Classification Accuracy.

Detection method	Deepfake Models	
	Deepfakes	CelebDF
UniDet w/ LC	66.60	11.01
Ours	79.63	11.70
Ours fintuned w/ CelebDF dataset	**87.29**	**50.34**

  - Fine-tuning with CelebDF datasets can increase the Deepfakes' performance to 87%.
  - Still show low performance on CelebDF dataset.

Our overall framework

- Inference



Deepfakes : 5,405 frames (= 2,707 real + 2,698 fake)

Qualitative results

Retrieved images from each stage are shown. Red-dotted box denotes missed deepfake.

ID 273, Real

Recall: 100% (22/22)

Recall: 85.7% (12/14)

Query

Stage 1

Stage 2

Deepfakes : 5,405 frames (= 2,707 real + 2,698 fake)

Qualitative results

Retrieved images from each stage are shown. Red-dotted box denotes missed deepfake.

ID 731, Real

Query

Recall: 100% (19/19)

Stage 1

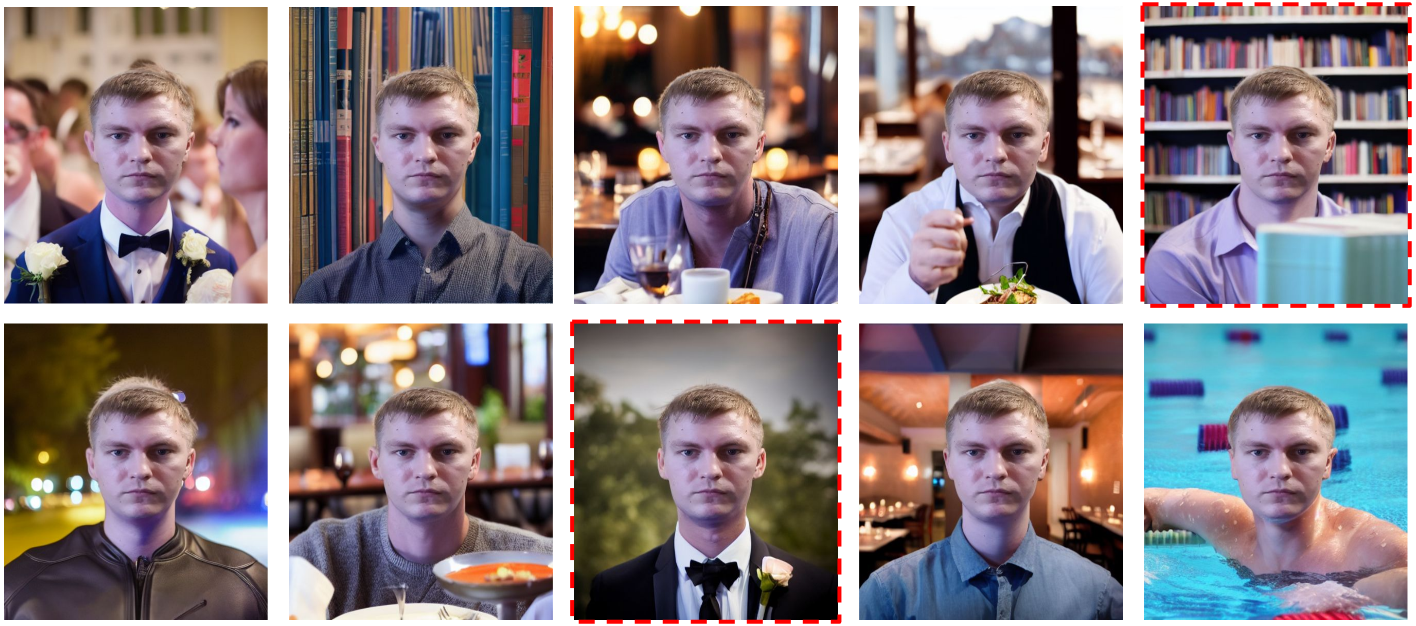
Recall: 80% (4/5)

Stage 2

Qualitative results

- We also check our model can detect modified images while preserving their identities.

Query



Recall: 80% (8/10)

Conclusion

- We first construct the **deepfake retrieval framework**.
  - Not just when identities change, but also when backgrounds change.
- Significant <u>performance improvement</u> compared to UniDet by **using text features**, especially on Deepfakes dataset.
- Limitation
  - The protocol of universal deepfake detection is based on ProGAN, but there is a lack of face images in this dataset, so we need a universal deepfake detection method that can overcome this problem.

Detection method	Generative Adversarial Networks						Denoising Diffusion Models			DALL-E	Deepfake Models		Avg.
	Pro-GAN	Cycle-GAN	Big-GAN	Style-GAN	Gau-GAN	Star-GAN	Glide	Guided	LDM		Deepfakes	CelebDF	
UniDet w/ LC	100.00	**98.50**	**94.50**	82.00	**99.50**	97.00	79.07	70.03	**94.19**	81.47	66.60	11.01	81.15
Ours	100.00	95.60	93.80	**95.25**	93.43	**99.15**	**92.88**	**84.3**	88.16	**91.50**	**79.63**	**11.70**	**93.22**

Roles

- Jumin
  - Implement CoOp
  - Generate PhotoGuard samples

- Suhyeon
  - Implement ArcFace
  - Implement inference framework

Q&A

Thank you.