세미나 발표 자료: FLIP

FLIP: Cross-domain Face Anti-spoofing with Language Guidance

> **Title :** FLIP: Cross-domain Face Anti-Spoofing with Language Guidance
>
> **Authors :** Koushik Srivatsan, Muzammal Naseer, Kartik Nandakumar
>
> **Venue :** IEEE International Conference on Computer Vision (ICCV)
>
> **Year :** 2023
>
> **Task :** Face Anti-Spoofing, Cross-domain generalization, Vision-Language Models
>
> **Keywords :** Face Anti-Spoofing, Cross-domain Generalization, Vision-Language Pre-training(VLP), CLIP, Language Guidance, Contrastive Learning, Vision Transformer(ViT)

안전한 Face Anti-Spoofing (FAS)을 위해서는 Cross-domain generalize performance가 필요한데 CLIP이라는 모델이 Vision task에서 generalization이 잘 된다고 하니, FAS에도 CLIP 모델을 활용하여 Robust Cross-domain generaizablity를 달성해보자.

1. Introduction - The Critical Need for Robust Face Anti-Spoofing (FAS)

왜 Face Anti-Spoofing (FAS)이 중요한가?

- **Face Recognition의 보편화**
  - 개인장치부터 공항 탑승 게이트 출입 통제, 금융 거래까지
  - 편리하고 비대면이라는 장점
- **취약점: Presentation Attacks**

    - 사진, 비디오 재생, 3D 마스크 등으로 신분을 위조하려는 시도
    - FAS 시스템의 보안을 무력화시키는 핵심 위협
- **결론:** 안전한 얼굴 인식 시스템을 위해 FAS는 필수이다

2. The Major Challenge: Cross-Domain Generalization

기존 FAS 방법들의 핵심적인 문제점 → 낮은 일반화 성능

- **Intra-domain vs Cross-domain**
    - **Intra-domain**: 학습 데이터와 동일한 환경(카메라, 조명, 위조 방식)에서는 높은 성능
    - **Cross-domain**: 학습에서 보지 못한 새로운 환경(카메라 센서, 조명 변화, 위조 도구, 환경 조건)에서는 성능이 급격히 저하
- **원인**
    - **Domain Gap**: 소스 도메인(학습 데이터)과 타겟 도메인(실제 환경) 간의 근본적인 분포 차이
    - **데이터 부족**: 실제 환경의 다양한 변화를 모두 커버할 만큼 충분한 학습 데이터 확보의 어려움
- **Prior Work Limitations**
    - **CNN 기반 방법**: 주로 지역적 특징(Local features)에 의존하여 전역적(global) 위조 패턴 파악에 한계
    - **ViT 기반 방법**: 장거리 의존성(long-range dependencies) 포착에 강점
        - 하지만 ImageNet 등 이미지 데이터로만 사전 학습된 ViT는 FAS 작업에 특화된 의미론적 이해가 부족
        - 추가적인 Adaptive modules나 도메인/공격 유형 정보를 요구하는 경우가 많아서 일반화 성능이 떨어짐

3. Core Idea: Leveraging Vision-Language Pre-training (VLP)

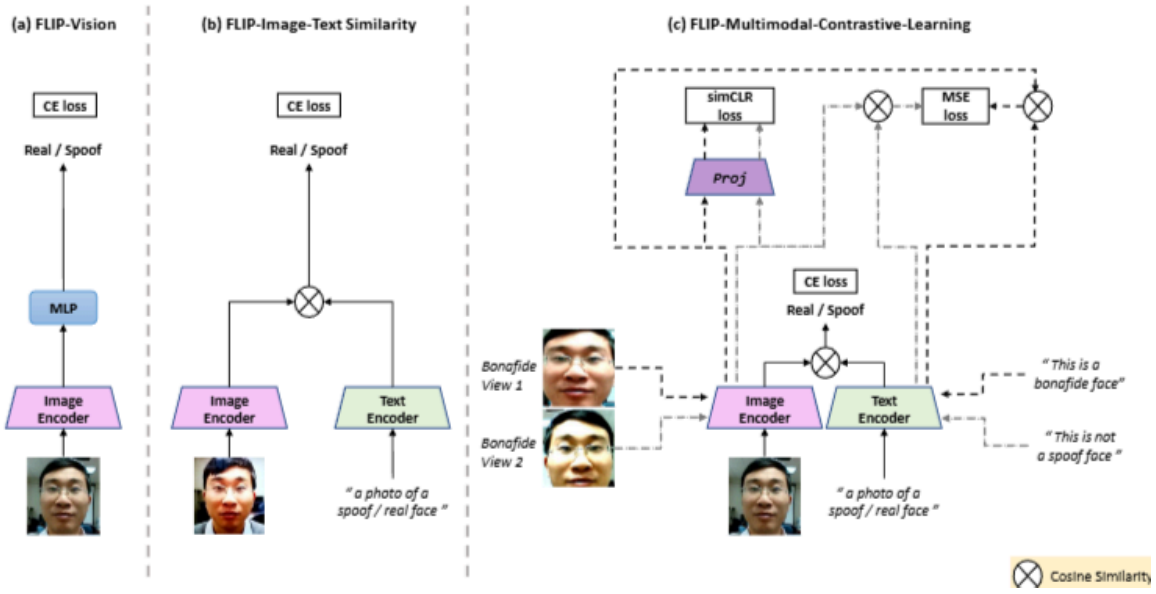
FLIP 핵심 아이디어: Vision-Language Pre-trained (VLP) 모델 활용

- **VLP 모델이란? (e.g. CLIP)**
  - 수백만 개의 **이미지-텍스트 쌍**으로 사전 학습된 모델. (CLIP은 4억개의 이미지 쌍을 학습)
  - 이미지와 텍스트를 **동일한 임베딩 공간**에 매핑하는 능력 학습
  - 결과적으로 시각적 정보와 언어적 의미를 동시에 이해하고 표현
- **FLIP 프레임워크의 목표**
  - VLP 모델의 이미지 인코더로 FAS 모델을 Fine-tuning하여 일반 이미지 사전 학습 능력을 향상
  - VLP 모델의 텍스트 인코더를 사용하여 FAS 성능 향상에 기여
  - VLP 모델을 FAS에 적용할 때, **Self-supervised learning 기법**을 추가하여 일반화 능력을 더욱 향상

4. Introducing FLIP: The Framework Overview

FLIP framework

- **Base model**: CLIP (Contrastive Language-Image Pre-training)
- 세 가지 Protocol 제안
  - **FLIP-Vision (FLIP-V)**: 사전 학습된 ViT의 이미지 인코더만 Fine-tuning 하고 MLP head를 추가.
  - **FLIP-Image-Text Similarity (FLIP-IT)**: FLIP-V에 텍스트 인코더를 추가하여 이미지 표현을 클래스별 텍스트 프롬프트 임베딩과 align 하여 분류
  - **FLIP-Multimodal-Contrastive-Learning (FLIP-MCL)**: FLIP-IP에 Multimodal contrastive learning을 추가하여 일반화 성능(Generalizability)을 극대화 → 최종 제안 기법

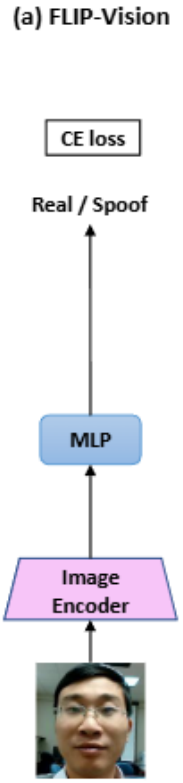


4.1. FLIP-Vision (FLIP-V)

- **아이디어**: CLIP의 사전 학습된 ViT 이미지 인코더를 FAS task에 맞게 fine-tune
- **구조**
  - 입력 이미지 I → CLIP ViT 이미지 인코더 V → 최종 클래스 토큰 c_K → ImageProj → Image representation x
  - x를 Multi-Layer Perceptro Classification head에 통과시켜 Real/Spoof 예측
- **학습**: 표준 Cross-Entropy Loss 사용

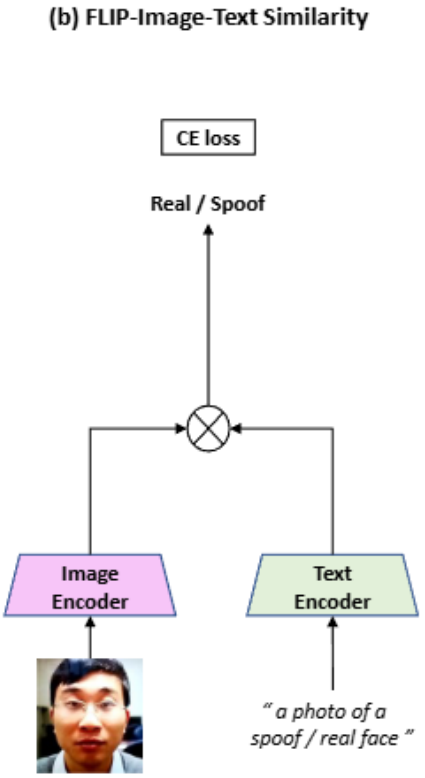
$$L_{ce} = CrossEntropy(MLP(x), y_{true})$$

- **핵심**: CLIP 모델의 '시각적 특징' 일반화 능력을 FAS에 활용



4.2. FLIP-Image-Text Similarity (FLIP-IT)

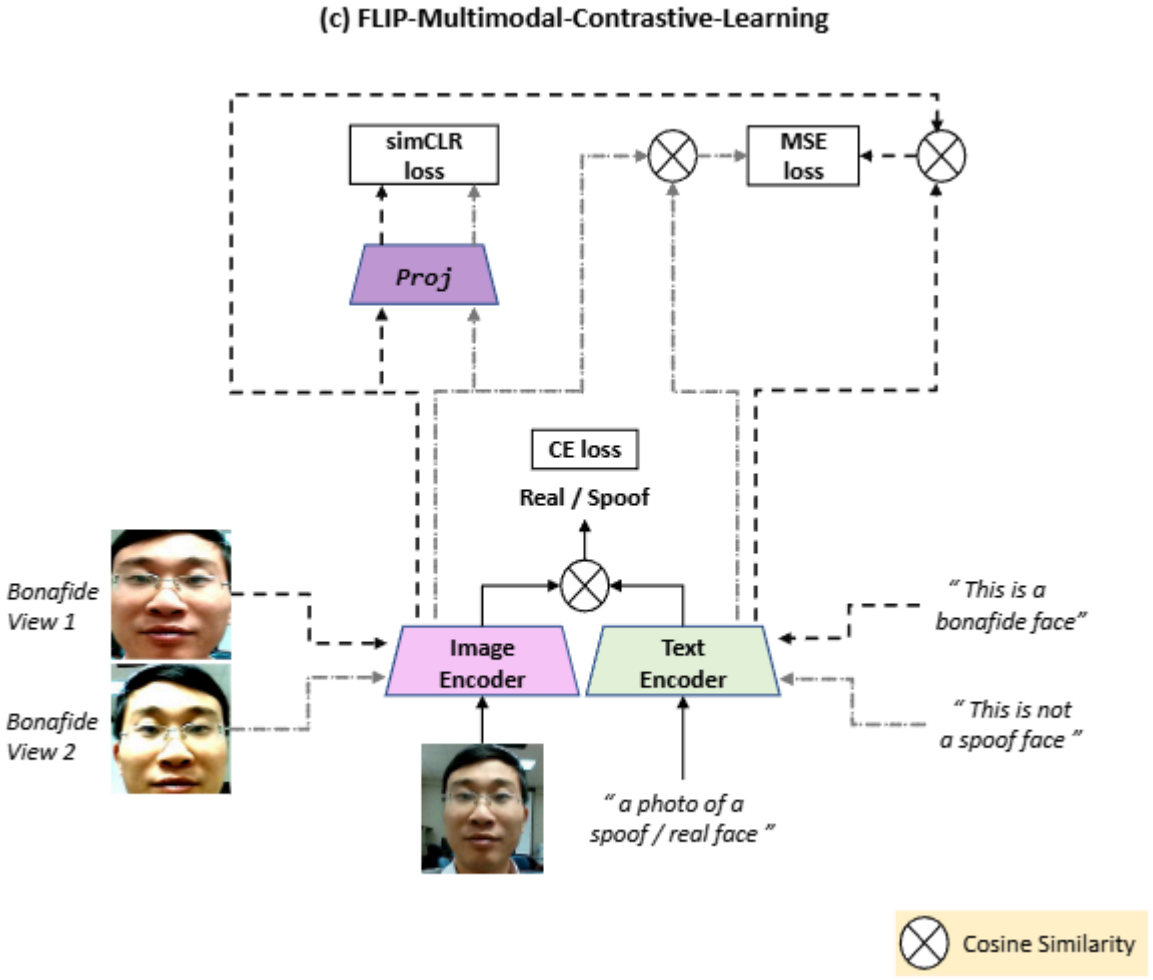
- **Language Guidance** 도입
- **핵심 아이디어**: Image representation을 **Text prompt**의 semantic representation과 정렬하여 분류
- **작동 방식**
  - 입력 이미지 I → ViT 인코더 V → 이미지 임베딩 x
  - 'Real' 및 'Spoof' 클래스를 설명하는 **텍스트 프롬프트** (e.g. "This is a real face", "This is a spoof face") → CLIP 텍스트 인코더 L → 텍스트 임베딩 z_r, z_s
  - **Ensemble**: 클래스당 여러 프롬프트를 사용하여 텍스트 임베딩의 평균($\bar{z}$)을 사용 → Robust Representation Learning
  - 이미지 임베딩 x와 텍스트 임베딩 $\bar{z}$ 간의 **Cosine similarity**를 계산하여 Logits으로 사용
  - Softmax 함수를 통해 확률 계산



$$p(\hat{y}|x) = \frac{\exp(\text{sim}(x, \bar{z}_{\hat{y}})/\tau)}{\exp(\text{sim}(x, \bar{z}_r)/\tau) + \exp(\text{sim}(x, \bar{z}_s)/\tau)}$$

- 이점
  - **의미론적 '안내(Grounding / Guidance)'**: 위조 공격의 미세한 특징(e.g.: 종이 질감, 화면 왜곡)을 텍스트 설명이라는 명확한 의미 기준에 연결
  - **도메인 간극 완화**: 텍스트는 이미지보다 도메인 변화에 덜 민감할 수 있으며, 이미지 특징을 텍스트 의미에 맞추면 일반화 성능 향상

4.3. FLIP-Multimodal-Contrastive-Learning (FLIP-MCL)



- **핵심 아이디어**: FLIP-IT에 **Self-Supervised Learning** 및 **Image-Text Similarity Consistency**을 추가하여 임베딩의 Robustness와 Domain-invariance 강화
- **전체 손실 함수**

$$L_{mcl} = \alpha L_{ce} + \beta L_{simCLR} + \gamma L_{mse}$$

  - L_{simCLR} **(Image-based Contrastive Loss)**
    - 동일 이미지 I에 서로 다른 변환(view) I_{v1}, I_{v2}을 적용
    - 각 view의 features x_{v1}, x_{v2}를 non-linear projection network H를 통해 h_{v1}, h_{v2}로 변환
    - h_{v1}, h_{v2} 간의 Contrastive learning으로 유사성 최대화

$$\boldsymbol{x}^{v_1} = \mathcal{V}(I^{v_1}), \quad \boldsymbol{x}^{v_2} = \mathcal{V}(I^{v_2})$$

$$\boldsymbol{h}_1 = \mathcal{H}(\boldsymbol{x}^{v_1}) \ , \ h_2 = \mathcal{H}(\boldsymbol{x}^{v_2}) \qquad \boldsymbol{h}_1, \boldsymbol{h}_2 \in \mathbb{R}^{d_h}.$$

$$L_{simCLR} = \mathrm{simCLR}(\boldsymbol{h}_1, \boldsymbol{h}_2)$$

- L_{mse} **(Image-Text View Consistency Loss)**
  - 두 개의 다른 Image view(x_{v1}, x_{v2})와 두 개의 다른 Text prompt view (z_{v1}, z_{v2})를 사용
  - 두 이미지-텍스트 쌍에서 계산된 Cosine similarity 점수($sim(x_{v1}, z_{v1})$과 $sim(x_{v2}, z_{v2})$) 간의 **평균 제곱 오차(MSE)** 최소화
  - 목적: 이미지-텍스트 뷰 쌍의 일관성을 강제. 도메인 간극 완화에 기여

$$L_{msc} = (sim(\boldsymbol{x}^{v_1}, \boldsymbol{z}^{v_1}) - sim(\boldsymbol{x}^{v_2}, \boldsymbol{z}^{v_2}))^2$$

5. Experiment Setup

- Datasets & DG Protocols
  - Protocol 1: MSU-MFSD(M), CASIA-MFSD(C), Replay Attack(I), OULU-NPU(O)
  - Protocol 2: WMCA(W), CASIA-CeFA(C), CASIA-SURF(S)
  - Protocol 3: 12개 Single-source to Single-target 시나리오
  - CelebA-Spoof: 보조 훈련 데이터
- Evaluation Metrics
  - Half Total Error Rate (HTER) ↓
  - Area Under ROC Curve (AUC) ↑
  - True Positive Rate at Fixed False Positive Rate (TPR@FPR=1%) ↑
- 비교 대상
  - SOTA Domain Generalization 방법
  - ViT 기반 FAS
  - **Zero-shot vs Five-shot**: 제안된 방법의 0-shot 성능으로 5-shot SOTA를 능가함을 강조

- Implementation
  - Image size: 224 × 224 × 3
  - Patch size: 16 × 16
  - Optimizer: Adam, Learning Rate: 10^{-6}, Weight decay: 10^{-6}
  - CLIP ViT base 사용
  - two-layer MLP head
  - Image representation $d_v = 768$
  - Vision-Language embedding dim $d_{vl} = 512$

6. Results

- Average HTER
- **Protocol 1**

Table 2. Evaluation of cross-domain performance in Protocol 1, between MSU-MFSD (M), CASIA-MFSD (C), Replay Attack (I) and OULU-NPU (O). We run each experiment 5 times under different seeds and report the mean HTER, AUC, and TPR@FPR=1%.

	Method	OCI → M			OMI → C			OCM → I			ICM → O			Avg.
		HTER	AUC	TPR@ FPR=1%	HTER	AUC	TPR@ FPR=1%	HTER	AUC	TPR@ FPR=1%	HTER	AUC	TPR@ FPR=1%	HTER
0-shot	MADDG (CVPR' 19) [38]	17.69	88.06	–	24.50	84.51	–	22.19	84.99	–	27.98	80.02	–	23.09
	MDDR (CVPR' 20) [44]	17.02	90.10	–	19.68	87.43	–	20.87	86.72	–	25.02	81.47	–	20.64
	NAS-FAS (TPAMI' 20) [53]	16.85	90.42	–	15.21	92.64	–	11.63	96.98	–	13.16	94.18	–	14.21
	RFMeta (AAAI' 20) [39]	13.89	93.98	–	20.27	88.16	–	17.30	90.48	–	16.45	91.16	–	16.97
	D^2AM (AAAI' 21) [6]	12.70	95.66	–	20.98	85.58	–	15.43	91.22	–	15.27	90.87	–	16.09
	DRDG (IJCAI' 21) [28]	12.43	95.81	–	19.05	88.79	–	15.56	91.79	–	15.63	91.75	–	15.66
	Self-DA (AAAI' 21) [46]	15.40	91.80	–	24.50	84.40	–	15.60	90.10	–	23.10	84.30	–	19.65
	ANRL (ACM MM' 21) [27]	10.83	96.75	–	17.85	89.26	–	16.03	91.04	–	15.67	91.90	–	15.09
	FGHV (AAAI' 21) [26]	9.17	96.92	–	12.47	93.47	–	16.29	90.11	–	13.58	93.55	–	12.87
	SSDG-R (CVPR' 20) [18]	7.38	97.17	–	10.44	95.94	–	11.71	96.59	–	15.61	91.54	–	11.28
	SSAN-R (CVPR' 22) [48]	6.67	98.75	–	10.00	96.67	–	8.88	96.79	–	13.72	93.63	–	9.80
	PatchNet (CVPR' 22) [42]	7.10	98.46	–	11.33	94.58	–	13.40	95.67	–	11.82	95.07	–	10.90
	GDA (ECCV' 22) [67]	9.20	98.00	–	12.20	93.00	–	10.00	96.00	–	14.40	92.60	–	11.45
0-shot	DiVT-M (WACV' 23) [23]	2.86	99.14	–	8.67	96.62	–	3.71	99.29	–	13.06	94.04	–	7.07
	ViT (ECCV' 22) [16]	**1.58**	**99.68**	**96.67**	5.70	98.91	88.57	9.25	97.15	51.54	7.47	98.42	69.30	6.00
5-shot	ViT (ECCV' 22) [16]	3.42	98.60	95.00	1.98	99.75	94.00	2.31	99.75	87.69	7.34	97.77	66.90	3.76
	ViTAF* (ECCV' 22) [16]	2.92	99.62	91.66	1.40	99.92	98.57	1.64	99.64	91.53	5.39	98.67	76.05	3.31
0-shot	FLIP-V	3.79	99.31	87.99	1.27	99.75	95.85	4.71	98.80	75.84	4.15	98.76	66.47	3.48
	FLIP-IT	5.27	98.41	79.33	**0.44**	**99.98**	99.86	**2.94**	**99.42**	**84.62**	3.61	99.15	84.76	3.06
	FLIP-MCL	4.95	98.11	74.67	0.54	**99.98**	**100.00**	4.25	99.07	**84.62**	**2.31**	**99.63**	**92.28**	**3.01**

  - SOTA 0-shot ViT: 6.00%
  - 5-shot ViTAF: 3.31%
  - FLIP-V: 3.48%
  - FLIP-IT: 3.06%
  - FLIP-MCL: 3.01%

- **Protocol 2**

Table 3. Evaluation of cross-domain performance in Protocol 2, between CASIA-SURF (S), CASIA-CeFA (C), and WMCA (W). We run each experiment 5 times under different seeds and report the mean HTER, AUC, and TPR@FPR=1%

	Method	CS → W			SW → C			CW → S			Avg.
		HTER	AUC	TPR@ FPR=1%	HTER	AUC	TPR@ FPR=1%	HTER	AUC	TPR@ FPR=1%	HTER
0-shot	ViT (ECCV' 22) [16]	7.98	97.97	73.61	11.13	95.46	47.59	13.35	94.13	49.97	10.82
5-shot	ViT (ECCV' 22) [16]	4.30	99.16	83.55	7.69	97.66	68.33	12.26	94.40	42.59	6.06
	ViTAF* (ECCV' 22) [16]	2.91	99.71	92.65	6.00	98.55	78.56	11.60	95.03	60.12	5.12
0-shot	FLIP-V	6.13	97.84	50.26	10.89	95.82	53.93	12.48	94.43	53.00	9.83
	FLIP-IT	4.89	98.65	59.14	10.04	96.48	59.4	15.68	91.83	43.27	10.2
	FLIP-MCL	**4.46**	**99.16**	**83.86**	**9.66**	**96.69**	**59.00**	**11.71**	**95.21**	**57.98**	**8.61**

- **Protocol 3: Challenging Single-Source to Target**

Table 4. Evaluation of cross-domain performance in Protocol 3, for all the 12 different combinations between MSU-MFSD (M), CASIA-MFSD (C), Replay Attack (I) and OULU-NPU (O). We run each experiment 5 times under different seeds and report the mean HTER.

	Method	C → I	C → M	C → O	I → C	I → M	I → O	M → C	M → I	M → O	O → C	O → I	O → M	Avg.
0-shot	ADDA (CVPR' 17) [40]	41.8	36.6	-	49.8	35.1	-	39.0	35.2	-	-	-	-	39.6
	DRCN (ECCV' 16) [12]	44.4	27.6	-	48.9	42.0	-	28.9	36.8	-	-	-	-	38.1
	DupGAN (CVPR' 18) [15]	42.4	33.4	-	46.5	36.2	-	27.1	35.4	-	-	-	-	36.8
	KSA (TIFS' 18) [21]	39.3	15.1	-	12.3	33.3	-	9.1	34.9	-	-	-	-	24.0
	DR-UDA (TIFS' 20) [45]	15.6	9.0	28.7	34.2	29.0	38.5	16.8	3.0	30.2	19.5	25.4	27.4	23.1
	MDDR (CVPR' 20) [44]	26.1	20.2	24.7	39.2	23.2	33.6	34.3	8.7	31.7	21.8	27.6	22.0	26.1
	ADA (ICB' 19) [43]	17.5	9.3	29.1	41.5	30.5	39.6	17.7	5.1	31.2	19.8	26.8	31.5	25.0
	USDAN-Un (PR' 21) [19]	16.0	9.2	-	30.2	25.8	-	13.3	3.4	-	-	-	-	16.3
	GDA (ECCV' 22) [67]	15.10	**5.8**	-	29.7	20.8	-	12.2	2.5	-	-	-	-	14.4
	CDFTN-L (AAAI' 23) [56]	**1.7**	8.1	29.9	11.9	9.6	29.9	8.8	**1.3**	25.6	19.1	5.8	6.3	13.2
0-shot	FLIP-V	15.08	13.73	12.34	4.30	9.68	7.87	0.56	3.96	4.79	2.09	5.01	6.00	7.12
	FLIP-IT	12.33	15.18	7.98	1.12	8.37	6.98	**0.19**	5.21	4.96	**0.16**	**4.27**	**5.63**	6.03
	FLIP-MCL	10.57	7.15	**3.91**	**0.68**	**7.22**	**4.22**	**0.19**	5.88	**3.95**	0.19	5.69	8.40	**4.84**

- **정리**
  - FLIP-V 만으로도 FAS 성능이 향상됨
  - FLIP-IT에서 Language guidance로 성능 추가 향상
  - FLIP-MCL, Multimodal contrastive learning으로 SOTA 성능 달성, 일부 지표에서는 0-shot 성능이 5-shot 성능을 능가함

7. Ablation Studies

- **Comparing ViT initialization methods for FAS**

Table 5. Comparing different ViT initialization methods for FAS. We use each initialization method with their default parameters and show the results for **Protocol 1**.

Method	OCI → M		OMI → C		OCM → I		ICM → O		Avg.
	HTER	AUC	HTER	AUC	HTER	AUC	HTER	AUC	HTER
Scratch	18.32	87.36	40.05	61.13	19.22	88.15	29.72	73.66	25.86
BeIT [1]	4.73	98.46	7.86	96.62	13.51	92.42	15.19	91.95	8.70
ImageNet [16]	**1.58**	**99.68**	5.70	98.91	9.25	97.15	7.47	98.42	6.00
CLIP (FLIP-V)	3.79	99.31	**1.27**	**99.75**	**4.71**	**98.80**	**4.15**	**98.76**	**3.48**

- **Impact of different text prompts**

Prompt No.	Real Prompts	Spoof Prompts
P1	This is an example of a real face	This is an example of a spoof face
P2	This is a bonafide face	This is an example of an attack face
P3	This is a real face	This is not a real face
P4	This is how a real face looks like	This is how a spoof face looks like
P5	A photo of a real face	A photo of a spoof face
P6	This is not a spoof face	A printout shown to be a spoof face

Table 1. Natural language descriptions (context prompts) of the real and spoof classes used to guide the FLIP-IT model.

Table 6. Impact of guidance with different text prompts (described in Table 1). We use FLIP-IT and show the results for **Protocol 1**.

Prompt	OCI → M		OMI → C		OCM → I		ICM → O		Avg.
	HTER	AUC	HTER	AUC	HTER	AUC	HTER	AUC	HTER
P1	6.00	98.17	0.54	99.97	3.60	99.19	3.47	99.24	3.40
P2	8.32	96.38	1.05	99.90	2.98	99.48	5.74	98.39	4.52
P3	**4.68**	**98.43**	**0.21**	**99.99**	4.30	99.06	4.07	99.02	3.31
P4	5.78	97.91	0.65	99.93	3.72	99.21	3.54	99.28	3.42
P5	6.48	98.37	0.46	99.96	**2.52**	**99.55**	3.24	99.30	3.17
P6	5.58	98.00	0.3	99.99	2.85	99.28	**3.03**	**99.46**	**2.94**
Ensemble	5.27	98.41	0.44	99.98	2.94	99.42	3.61	99.15	3.06

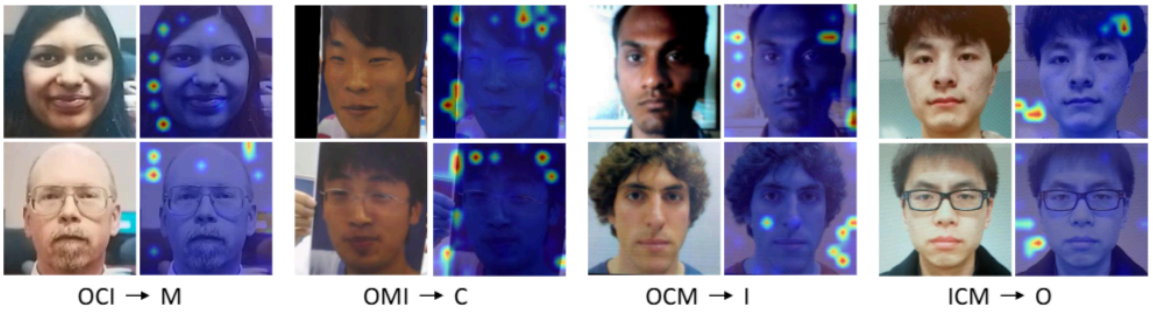
- **Contribution of different loss terms**

Table 7. Average HTER performance under different loss weights for Protocol 1. $L_{mcl} = \alpha L_{ce} + \beta L_{simCLR} + \gamma L_{mse}$

(α, β, γ)	(1,1,1)	(1,1,0)	(1,0,1)	(1,2,2)	(1,5,5)
HTER	3.01	3.15	3.47	3.20	3.67

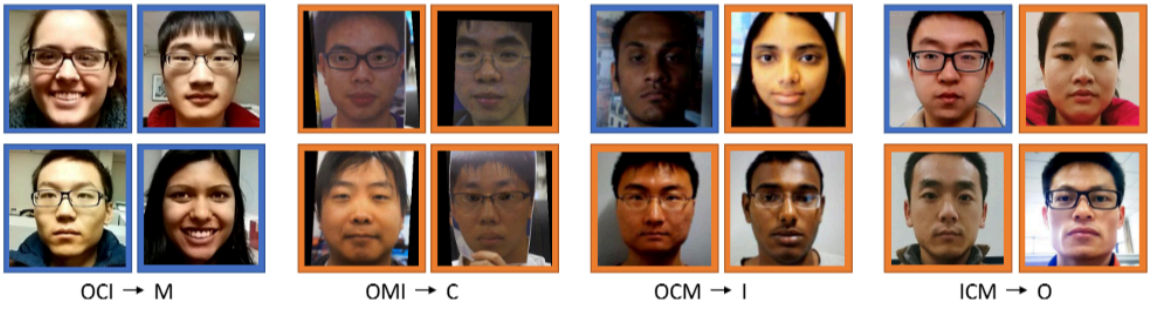
8. Visualization

- Attention Maps (Spoof Images)



  - 모델이 위조 특징(e.g. 종이 질감, 테두리, 모아레 패턴, 옷 주름, 화면 가장자리)을 효과적으로 감지하고 있음을 보여줌

- Mis-classified Examples



- **Real → Spoof**: 낮은 해상도, 조명 변화, 배경 텍스처 등 실제 얼굴 특징이 위조처럼 오인되는 경우
- **Spoof → Real**: 고해상도 위조 샘플, 실제와 구분하기 어려운 정교한 위조 공격
- 일부 어려운 케이스가 존재하지만 전반적으로 모델이 복잡한 도메인 변화에도 잘 대처하고 있음
  - OCI → M, no false positive cases
  - For OCM → I, 실제 샘플의 0.62%만 잘못 분류됨을 관찰
  - For ICM → O, 실제 샘플의 0.2%가 Spoofing으로 오분류

9. Conclusion

- **강점**
  - 탁월한 Cross-Domain generalize performance
  - VLP 모델 활용하여 간결한 접근 방식
  - Text prompt의 효과적 활용
  - 견고성 강화: Multimodal contrastive learning으로 다양한 환경에 대한 Robustness 확보
- **한계점**
  - 계산 비용 증가: Text encoder의 활용으로 학습 및 추론 시 추가적인 연산 필요
  - VLP 모델 의존성: CLIP 모델의 품질 및 일반화 성능에 따라 FAS 성능이 결정됨
  - 효과적인 프롬프트 구성을 위한 Domain/Attack 특성에 대한 이해 필요
- **결론**

- VLP 모델(CLIP)을 FAS에 직접 적용하는 것은 매우 효과적이며, 특히 Language-guidance와 multimodal contrastive learning을 결합한 FLIP 프레임워크는 SOTA Cross-domain generalization performance를 달성함
- 시각적 특징을 언어적 의미와 결합하여 FAS의 복잡한 문제를 해결할 잠재력이 있음을 보여줌

- **향후 연구 방향**
  - **다른 VLP 모델 탐색**: BERT, ALIGN, BLIP 등 다양한 VLP 모델로 일반화 가능성 검증
  - **Prompt Learning**: 고정된 프롬프트 대신 학습 가능한 프롬프트나 dynamic 프롬프트 생성 방식 연구
  - **효율성 개선**: Text Encoder의 연산 부담을 줄이기 위한 경량화, Knowledge distillation 등의 연구
  - 실시간 추론 속도 향상 및 다양한 실제 환경 데이터셋에서의 검증

10. Q&A