

Domain Shift Preservation for Zero Shot Domain Adaptation

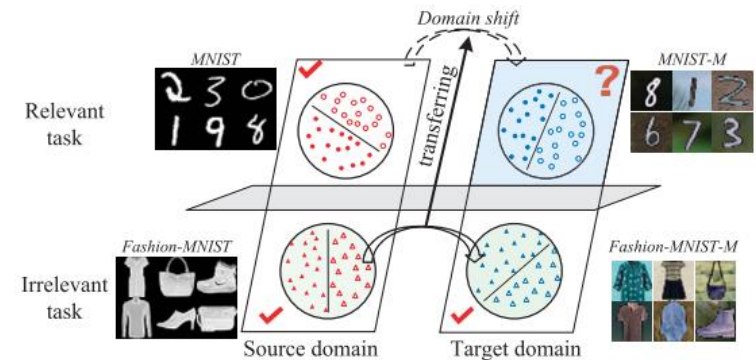
Wang, J., Cheng, M. M., & Jiang, J. (2021). Domain shift preservation for zero-shot domain adaptation. IEEE Transactions on Image Processing, 30, 5505-5517.

Introduction

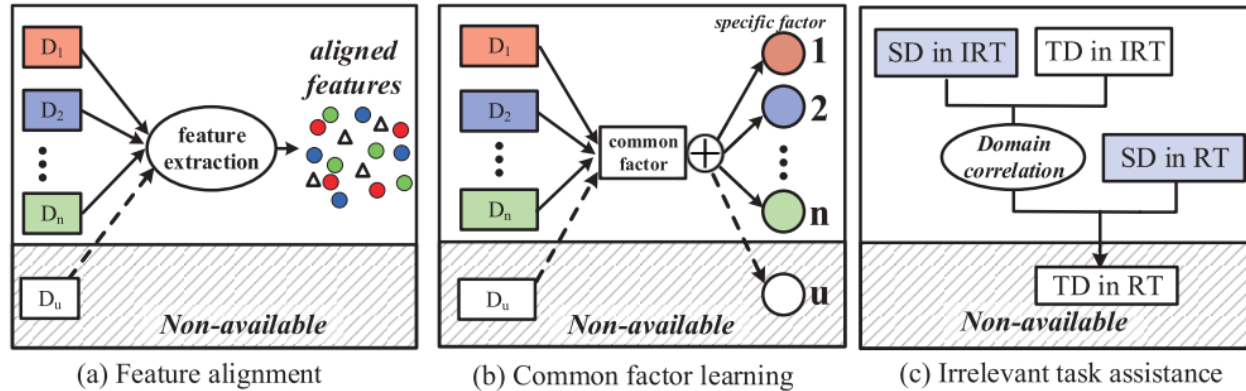
- Domain Adaptation은 학습 시 사용하는 타겟 도메인에 대한 정보에 따라 1) supervised 2) semi-supervised 3) Unsupervised 4) Zero-shot 이렇게 4가지로 나뉨
- ZSDA는 타겟 도메인에 대한 정보가 전혀 없는 상태를 뜻하며 ‘도메인 일반화’ 라고도 불림
- 도메인에 관계없는 공통된 속성 학습, 도메인 이동(Domain Shift)를 최소화, 도메인 정렬 등 다양한 방법들이 사용되지만 이러한 방법들은 비교할 다른 소스 도메인이 없는 경우 싱글 소스 도메인 상황에서는 사용할 수 없다는 단점을 지님
- 본 논문은 실제로 해결하고 싶은 목표에 관련된 데이터셋(RT / 소스 데이터만 사용 가능) 과 목표와 전혀 관계 없는 데이터셋(IRT / 소스 & 타겟 모두 사용 가능)를 사용하여 전혀 정보가 없는 목표의 타겟 데이터에 대해 강건한 성능을 만드는 방법에 대해 연구

DEPENDENT ON WHETHER LABELED AND UNLABELED DATA ARE AVAILABLE IN THE TARGET DOMAIN (T-DOMAIN), DOMAIN ADAPTATION TASKS CAN BE CLASSIFIED INTO FOUR CATEGORIES: SUPERVISED, SEMI-SUPERVISED, UNSUPERVISED AND ZSDA

T-Domain	Superv.	Semi-S.	Unsuperv.	ZSDA
Labeled	✓	✓	×	×
Unlabeled	×	✓	✓	×



Related Work – Zero Shot Domain Adaptation

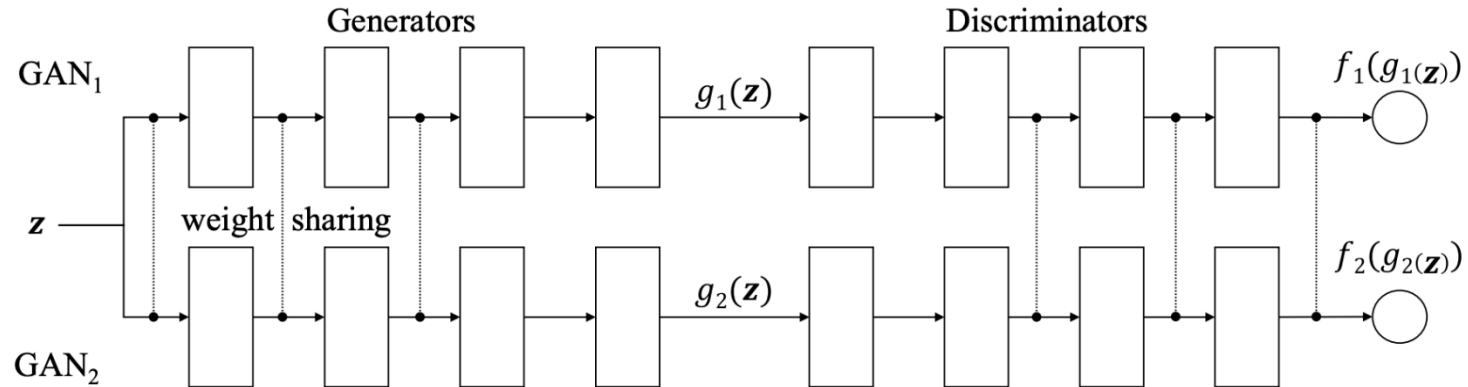


- ZSDA에는 크게 3가지 방법이 존재

- 1) 여러 소스 도메인($\{D_1, D_2, \dots, D_n\}$)에서 정렬된 특징을 생성하는 특징 추출 과정을 학습하고, 동일한 과정을 타겟 도메인(D_u)에 적용
Ex) 각각 다른 언어(영어, 프랑스어, 독일어 등)로 작성된 텍스트에서 주제를 분석하는 공통 특징 추출 과정을 학습한 후, 처음 접하는 도메인인 새로운 언어(예: 한국어)의 텍스트에 적용하는 것
- 2) 각 도메인을 공통 요소와 도메인 특정 요소로 분해하고, 공통 요소를 대상 도메인에 적용
Ex) MNIST와 Color MNIST에서 각각 공통 요소(숫자의 일반적인 특징)와 도메인 특정 요소(색깔, 데이터 셋의 특징)를 학습하고, 이를 알 수 없는 새로운 도메인(예: 다른 숫자 이미지)에 적용하는 것
- 3) 소스 도메인과 타겟 도메인 데이터가 모두 사용 가능한 데이터셋에서 도메인 상관 관계를 학습하고 목표로 하는 task에 전달
Ex) 특정 상황(RT)에서 수집한 얼굴 표정(소스 도메인)과 감정 상태(대상 도메인) 사이의 관계를 학습하고, 이를 새로운 상황(IRT)에서의 얼굴 표정과 감정 상태 사이의 관계 예측에 적용하는 것

Related Work – CoGAN^[*]

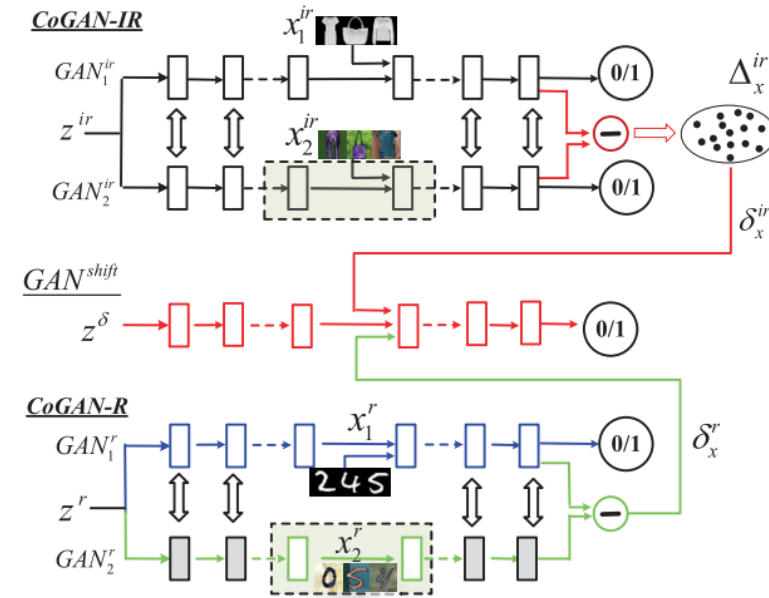
Liu, M. Y., & Tuzel, O. (2016). Coupled generative adversarial networks. *Advances in neural information processing systems*, 29.



- Multi-domain image의 Joint Distribution을 학습하는 문제를 해결하는 방법을 제안
- 두 개의 연결된 GAN을 도입하는데 서로 다른 데이터 분포를 모델링하지만, 두 네트워크의 일부는 가중치를 공유
- 공유된 가중치는 두 데이터 분포에 대해 공통된 표현을 학습
Ex) 두 개의 도메인이 흑백 이미지와 이에 대응하는 컬러 이미지라면 공유된 weight에서는 class 정보(low level), 후반부 layer에서는 컬러-흑백(high level semantic) 학습

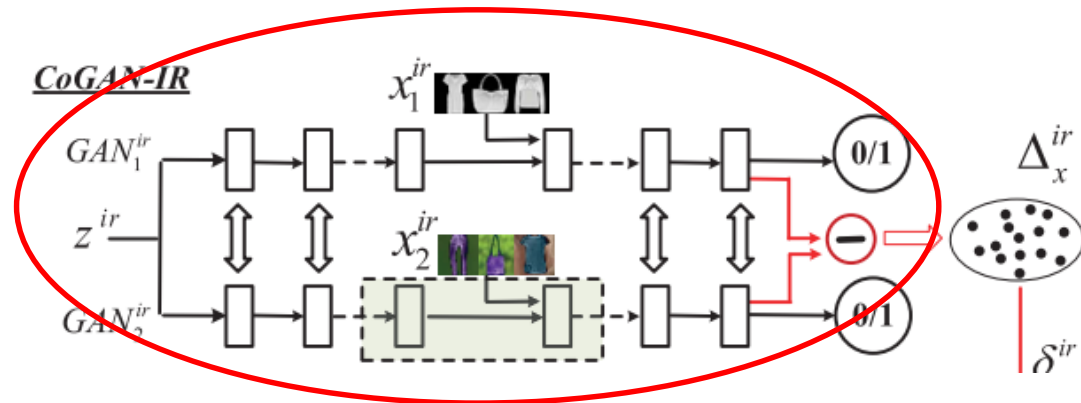
Related Work - PROPOSED METHOD

- ZSDA(Zero-Shot Domain Adaptation) 문제를 해결하기 위해, 두 개의 연결된 CoGANs를 사용하여 도메인 쌍 사이의 상관관계를 포착 & 하나의 GAN을 사용하여 도메인 변화를 모델링하는 방법 제안
- 제안하는 모델은 크게 3 부분으로 나뉨
 1. 도메인에 해당하는 특징적인 부분과 도메인이 변하더라도 변하지 않는 high level의 개념을 모델링하기 위해 IRT(목표X)에 대한 CoGAN-IR을 학습
 2. GAN 모델을 사용하여 IRT pair 데이터셋의 도메인 차이에 대한 정보를 학습
 3. 2에서 학습된 도메인 변화에 대한 정보를 RT(목표) 데이터를 학습하는 CoGAN-R에 전달 & 목표 데이터의 소스 데이터를 통해 CoGAN-R 학습



Related Work - PROPOSED METHOD

- IRT 데이터셋을 활용해 CoGAN 학습
- 이때 사용되는 데이터셋은 서로 같은 클래스를 가지며 색 차이와 같은 도메인 변화만 있는 pair dataset
- 해당 식은 GAN과 같은 실제 데이터와 가짜 데이터를 구별하도록 하는 min-max게임과 관련된 식
- E : 기댓값, z : 노이즈 벡터의 샘플, $f(x)$: 판별기가 IRT 소스 인스턴스에 대해 실제로 분류할 확률, $f(g(z))$: 판별기 f 가 생성기 g 가 만든 가짜 샘플을 실제로 분류할 확률
- $E_x[\log f(x)]$, 는 판별기가 실제 데이터를 실제로라고 분류하는 능력을 최대화
- $E_z[\log(1 - f(G(z)))]$: 판별기가 가짜 데이터를 가짜로 분류하는 능력을 최대화



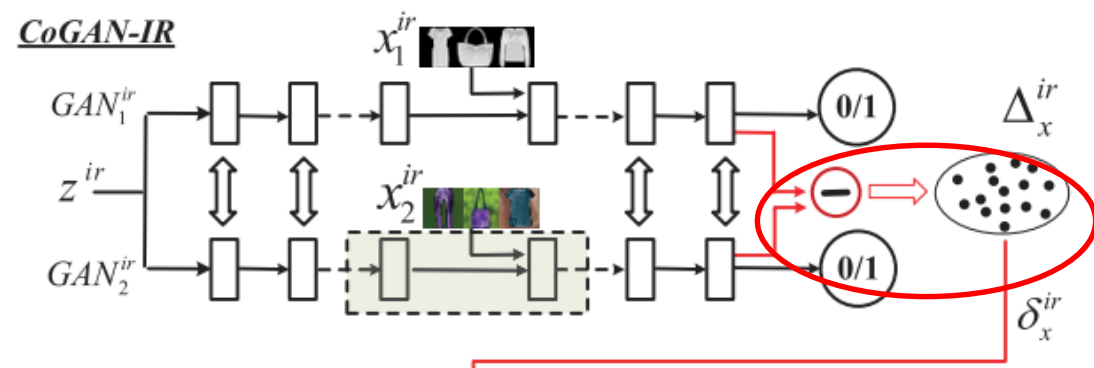
Training Procedure

Stage 1 Train *CoGAN-IR* to capture the joint distribution of (x_s^{ir}, x_t^{ir}) by optimizing Eq. (3).

$$\begin{aligned} \max_{g_1^{ir}, g_2^{ir}} \min_{f_1^{ir}, f_2^{ir}} V(f_1^{ir}, f_2^{ir}, g_1^{ir}, g_2^{ir}) \equiv \\ E_{x_s^{ir} \sim p_{x_s^{ir}}} [-\log f_1^{ir}(x_s^{ir})] + E_{z^{ir} \sim p_{z^{ir}}} [-\log(1 - f_1^{ir}(g_1^{ir}(z^{ir})))] \\ + E_{x_t^{ir} \sim p_{x_t^{ir}}} [-\log f_2^{ir}(x_t^{ir})] + E_{z^{ir} \sim p_{z^{ir}}} [-\log(1 - f_2^{ir}(g_2^{ir}(z^{ir})))], \end{aligned}$$

Related Work - PROPOSED METHOD

- CoGAN-IR이 학습이 완료 되었다면 학습한 생성 모델을 통해 fake pair images 생성
- 생성한 fake pair images 에 대한 차이값 델타를 구함
- 이때 델타는 단순 element-wise 인 판별하기 직전의 layer 값의 차
- 이를 통해 IRT 데이터를 통해서 배운 도메인의 차이 정보가 델타에 저장되는 효과를 가짐



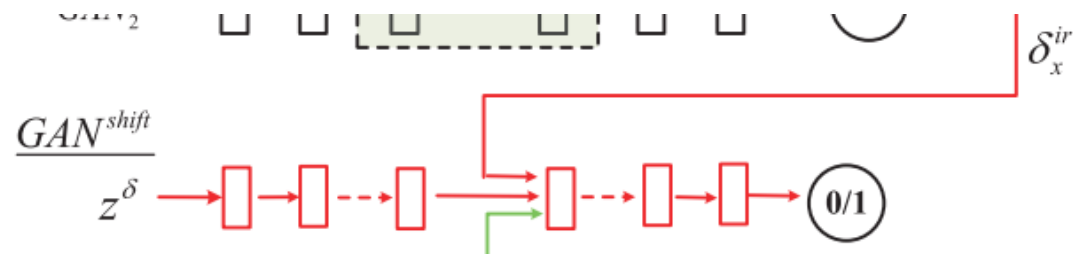
Stage 2 Model the domain shift explicitly:

2.1 Generate a set of *paired samples* $(\tilde{x}_s^{ir}, \tilde{x}_t^{ir})$ with *CoGAN-IR*;

2.2 Extract the representations $R(\tilde{x}_s^{ir})$ and $R(\tilde{x}_t^{ir})$ with *CoGAN-IR* and calculate their element-wise difference as $\delta_x^{ir} = R(\tilde{x}_s^{ir}) \ominus R(\tilde{x}_t^{ir})$;

Related Work - PROPOSED METHOD

- 도메인 간 차이를 모델링 하기 위해 GANshift라는 GAN 모델 학습
- GANshift 는 도메인 간 차이를 CoGAN-IR에서 representation space로 임베딩한 값들의 차이인 델타를 이용
- E : 기댓값, z : 랜덤 노이즈 벡터의 샘플, $f(x)$: 판별기가 도메인끼리의 차이 정보 인스턴스에 대해 실제로 분류할 확률, $f(g(z))$: 판별기 f 가 생성기 g 가 만든 가짜 샘플을 실제로 분류할 확률
- 앞선 CoGAN 처럼 서로 경쟁하여 GANshift 모델 학습
- 이를 통해 GANshift는 한 도메인에서 다른 도메인으로 지식을 전이하는데 도움을 줌

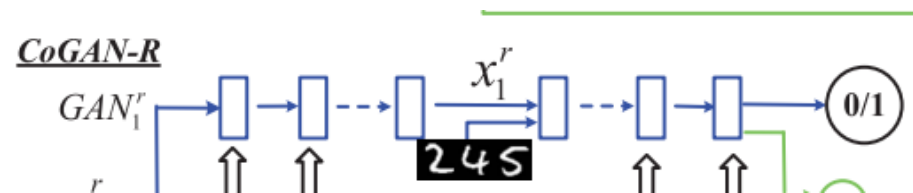


2.3 Train GAN^{shift} to capture the distribution of δ_x^{ir} by optimizing (4).

$$\max_{g^\delta} \min_{f^\delta} V(f^\delta, g^\delta) \equiv E_{\delta_x^{ir} \sim p_{\delta_x^{ir}}} [-\log f(\delta_x^{ir})] + E_{z^\delta \sim p_{z^\delta}} [-\log(1 - f^\delta(g^\delta(z^\delta)))], \quad (4)$$

Related Work - PROPOSED METHOD

- 실제 목표로 하고 있는 데이터(RT)에 대해 Zero-shot Domain Adaptation을 수행하기 위해 RT 소스 데이터만을 활용하여 GAN 모델 학습
- 앞선 설명과 같은 방식으로 일반 GAN 모델처럼 일단 source dataset에 대해 GAN 모델 학습



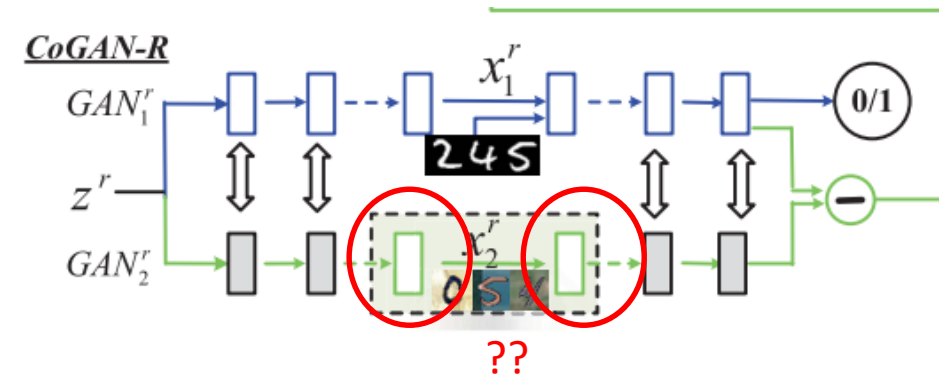
Stage 3. Train *CoGAN-R* to capture the joint distribution of (x_s^r, x_t^r) :

3.1 Train GAN_1^r to capture the distribution of x_s^r ;

$$\begin{aligned}
 & \max_{g_2^{r+}} \min_{f_2^{r+}} V(f_2^{r+}, g_2^{r+}) \\
 & \equiv E_{x_t^r \sim p_{x_t^r}} [-\log f_2^{r+}(x_t^r)] \\
 & \quad + E_{z^{r+} \sim p_{z^{r+}}} [-\log(1 - f_2^{r+}(g_2^{r+}(z^{r+})))] . \quad (5)
 \end{aligned}$$

Related Work - PROPOSED METHOD

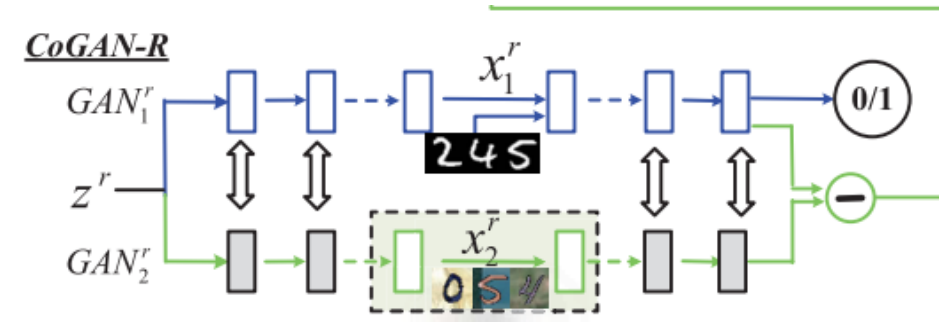
- 실제 목표로 하고 있는 데이터(RT)에 대해 Zero-shot Domain Adaptation을 수행하기 위해 RT 소스 데이터와 RT 타겟 데이터 간의 CoGAN-R 생성이 목적
- Shared weight는 3.1에서 학습한 weight를 사용
- 하지만 타겟 도메인 정보를 학습하는 GANr2 Non Shared weight에 대한 정보 존재 X
- 그렇기 때문에 일단 앞선 IRT 데이터셋의 target GAN의 lower layer를 초기 weight로 설정



3.2 Initialize the parameters of the non-shared layers in GAN_2^r with their counterparts in GAN_2^{ir} ;

Related Work - PROPOSED METHOD

- 초기 weight에서 목표로 하는 데이터셋의 타겟 도메인에 대해 맞추기 위해 fine-tuning 진행
- 목표로 하는 데이터셋의 클래스가 앞서 구한 IRT의 델타만큼의 차이를 가지기 위해 weight를 조정하는 것이 목적
- 이때 Fine-tuning은 Non-shared weight만 진행이 되며 loss 함수는 RT 데이터셋 target dataset에 대한 GAN 모델의 판별기가 가짜 데이터를 가짜로 분류하는 능력을 최대화하는 것을 목표로 계산됨
- 이를 통해 목표로 하는 데이터셋의 Unseen domain에 대해서 보다 좋은 성능을 낼 수 있도록 weight 수정



3.3 Fix the shared layers in GAN_2^r and train the non-shared layers by optimizing Eq. (6).

$$\min_{g_2^r, f_2^r} V(g_2^r, f_2^r) \equiv E_{z^r \sim p_{z^r}} [-\log(1 - f^\delta(\delta_x^r))]. \quad (6)$$

Related Work - PROPOSED METHOD

- 테스트 단계에서 목표 데이터셋의 타겟 도메인에 대한 분류 진행
- 목표 데이터셋의 소스 도메인을 이용해 classifier 생성
- 앞선 단계에서 학습한 CoGAN-R을 이용해 목표 데이터셋에 대한 fake pair images 생성
- 목표 데이터셋 소스 데이터의 label을 앞서 학습한 classifier를 통해 예측하고 이를 타겟 도메인의 샘플에 할당
- 생성된 가짜 타겟 도메인 샘플에 대해 classifier 학습 / 이때, $h_s(\tilde{x}_s^r) = h_t(\tilde{x}_t^r)$. 이어야만 한다는 regularization이 추가되며 이는 목표 소스 데이터셋과 목표 타겟 데이터셋이 class에 대한 공통된 특징을 이용해 분류되기 위한 제약식이다.
- 이를 통해 원본 도메인에서 학습된 모델이 한번도 경험하지 못한 타겟 도메인에 대해 강건한 성능을 가질 수 있도록 분류기 생성

Testing Procedure

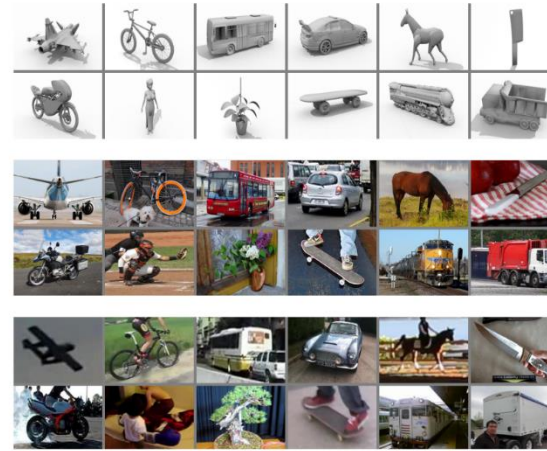
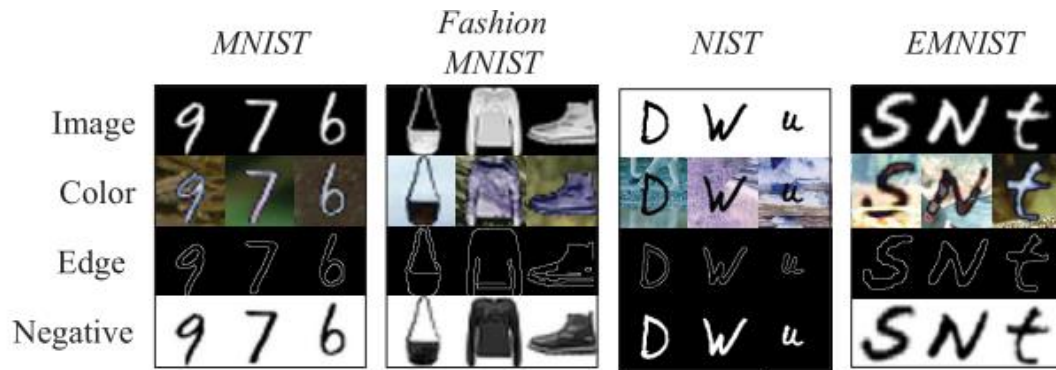
Stage 1 Train the classifier $h_s(\cdot)$ for X_s^r in the source domain;

Stage 2 Generate a set of *paired samples* $(\tilde{x}_s^r, \tilde{x}_t^r)$ with CoGAN-R;

Stage 3 Predict the label of \tilde{x}_s^r and assign it to \tilde{x}_t^r for each sample pair $(\tilde{x}_s^r, \tilde{x}_t^r)$;

Stage 4 Train a classifier $h_t(\cdot)$ for \tilde{x}_t^r with the constraint $h_s(\tilde{x}_s^r) = h_t(\tilde{x}_t^r)$.

Experiment



- 실험은 “MNIST, Fashion MNIST, NIST, EMNIST”, “VisDA2017”, “DomainHome” 데이터셋을 활용해 실험 진행
- 이때 도메인끼리의 차이는 gray domain (G-dom), color domain (C-dom), edge domain (E-dom / canny detector transforms), negative domain (N-dom / $255 - I$) 이며 비슷한 데이터셋들이 IRT 데이터셋으로 취급 되는 것은 아님

Experiment

TABLE II

THE ACCURACY (%) ON D_M , D_F , D_N AND D_E (BOLD-RED INDICATES THE BEST AND BOLD-BLACK INDICATES THE 2nd BEST)

(Source, Target)	RT	MNIST (D_M)			Fashion-MNIST (D_F)			NIST (D_N)		EMNIST (D_E)	
	IRT	D_F	D_N	D_E	D_M	D_N	D_E	D_M	D_F	D_M	D_F
(G-dom, C-dom)	DIVA [23]	75.9±2.8	86.3±3.0	87.1±1.4	52.5±5.5	57.3±1.1	62.4±4.6	44.1±3.0	39.5±4.8	72.4±3.9	50.7±1.1
	JiGen [24]	61.6±2.5	78.4±4.1	73.1±3.4	47.7±2.6	49.6±4.5	60.9±2.1	39.2±4.1	37.0±1.7	58.9±4.9	49.9±3.5
	MATE [22]	78.6±2.9	82.7±6.5	89.7±2.4	50.5±3.4	54.6±5.7	62.1±5.2	43.5±5.8	46.1±2.2	75.7±5.4	51.4±7.0
	Epi-FCR [25]	74.3±5.5	86.6±3.5	88.8±3.5	57.4±1.2	56.8±5.4	59.9±4.7	38.6±2.0	43.6±3.7	71.7±4.3	54.1±4.3
	MMD-AAE [26]	58.8±4.1	69.3±1.9	69.6±5.9	39.3±3.0	42.5±5.2	49.8±3.7	36.2±5.8	40.0±3.6	63.0±3.6	44.0±5.4
	CoCoGAN [16]	78.0±0.9	79.2±3.0	95.6±1.2	56.8±1.3	56.7±0.9	66.8±1.5	41.0±1.1	44.9±1.7	75.0±0.9	54.8±0.9
	ZDDA+PD [15]	73.4±1.9	89.8±2.4	93.5±1.5	50.7±0.9	43.9±1.1	64.7±1.6	34.8±2.8	40.3±2.8	69.2±3.0	56.7±3.7
	ZDDA+DC [15]	69.5±1.4	83.9±2.3	86.3±2.7	45.7±2.8	41.2±2.0	59.6±2.1	29.3±1.5	37.4±1.7	52.7±2.6	40.8±4.1
	Ours	82.9±1.3	94.4±2.8	95.3±1.4	59.1±0.8	58.6±2.2	63.1±2.7	46.4±2.9	46.5±4.0	73.6±2.0	58.3±1.6
(G-dom, E-dom)	DIVA [23]	74.7±7.3	87.2±7.1	89.5±3.3	61.3±3.9	58.4±3.6	67.5±4.3	44.0±2.5	45.3±3.7	75.8±2.9	54.7±6.8
	JiGen [24]	68.7±2.4	78.0±5.5	72.4±7.2	59.4±4.9	46.0±5.8	59.1±2.1	42.8±2.4	46.6±7.4	64.2±3.6	54.6±3.8
	MATE [22]	76.8±2.3	89.9±6.1	88.6±4.3	62.4±4.0	56.3±1.5	67.6±2.6	47.3±4.2	44.4±2.6	69.4±7.1	56.4±4.1
	Epi-FCR [25]	74.7±7.3	91.9±2.5	87.6±5.0	58.7±3.8	56.5±3.8	65.6±4.0	48.3±3.5	48.5±3.9	73.2±4.4	57.7±4.3
	MMD-AAE [26]	57.7±3.3	72.5±2.3	74.7±3.8	52.7±4.7	50.2±4.3	59.4±3.7	43.5±3.2	36.6±3.3	65.7±4.1	47.3±5.8
	CoCoGAN [16]	79.6±1.2	94.9±0.5	95.4±0.7	61.5±1.1	57.5±1.4	71.0±0.8	48.0±0.8	36.3±1.2	77.9±1.3	58.6±1.4
	ZDDA+PD [15]	72.5±4.8	91.5±1.3	93.2±2.7	54.1±4.3	54.0±5.9	65.8±4.9	42.3±1.8	44.4±3.9	73.6±1.9	60.7±3.9
	ZDDA+DC [15]	68.4±1.6	84.7±2.0	86.4±1.3	47.7±3.2	51.2±1.9	57.7±1.3	41.2±3.0	42.2±2.1	66.4±2.7	53.2±1.6
	Ours	84.7±2.4	93.9±1.6	95.5±2.1	64.8±1.0	60.4±2.3	73.3±1.8	50.2±2.0	47.6±1.4	79.2±1.4	64.7±2.6
(G-dom, N-dom)	DIVA [23]	76.0±4.7	76.3±2.9	88.2±3.3	62.5±5.2	55.3±4.7	70.6±4.0	46.8±3.3	51.1±6.2	77.0±5.1	61.0±3.8
	JiGen [24]	62.5±4.1	72.9±3.1	79.4±5.2	54.7±3.8	47.7±4.7	58.9±5.4	36.5±7.9	41.5±4.7	67.9±4.4	55.1±3.1
	MATE [22]	76.1±4.8	81.5±5.1	85.1±3.2	69.2±3.7	53.2±5.5	66.3±3.9	48.9±2.0	54.5±6.4	75.9±6.6	59.8±5.3
	Epi-FCR [25]	77.8±5.8	84.3±1.1	86.1±3.8	68.0±3.0	54.8±3.8	67.7±5.7	45.3±7.9	55.5±3.8	78.7±5.6	63.4±2.3
	MMD-AAE [26]	83.1±4.9	43.7±4.9	64.4±7.1	46.2±5.5	20.8±2.6	59.2±7.1	43.8±5.2	45.7±8.5	69.4±0.9	48.2±5.4
	CoCoGAN [16]	80.3±1.1	87.5±0.7	93.1±1.3	66.0±1.1	52.2±0.8	69.3±0.6	45.7±0.5	53.8±1.0	81.1±1.1	56.5±1.1
	ZDDA+PD [15]	77.9±4.8	88.2±2.8	90.5±0.8	61.4±3.4	47.4±2.4	62.7±3.9	37.8±2.5	46.7±3.0	79.2±2.8	53.4±2.9
	ZDDA+DC [15]	67.2±0.8	76.3±2.9	84.2±1.8	56.3±1.6	47.7±2.4	56.1±3.7	32.6±2.5	30.6±1.6	62.1±1.5	43.4±3.5
	Ours	83.6±1.7	89.1±1.6	94.2±2.4	71.1±1.3	58.0±2.3	70.3±1.9	49.6±3.0	57.4±1.6	82.1±2.5	65.2±2.2
(C-dom, G-dom)	DIVA [23]	65.8±4.5	91.1±2.0	92.7±3.2	60.8±3.7	46.0±4.8	68.4±1.1	53.5±4.4	53.1±5.3	78.3±5.5	64.1±4.3
	JiGen [24]	56.9±6.5	75.7±4.0	80.0±2.5	54.2±2.3	32.2±3.1	51.0±3.9	40.1±4.1	49.6±2.5	68.8±3.0	52.4±6.2
	MATE [22]	69.9±5.8	93.3±2.2	86.5±3.8	62.6±3.3	55.5±8.3	71.1±6.4	48.1±4.4	66.1±1.6	64.7±3.0	76.2±2.2
	Epi-FCR [25]	70.8±6.1	89.2±3.8	90.8±4.9	66.6±3.0	52.8±4.5	66.3±1.7	51.7±2.2	55.8±3.8	83.0±2.3	66.6±6.5
	MMD-AAE [26]	55.0±6.8	73.9±2.2	75.1±4.3	40.0±7.1	41.2±5.6	63.9±0.8	38.2±4.3	38.1±6.7	62.2±5.4	63.8±4.7
	CoCoGAN [16]	73.2±1.3	89.6±0.6	94.7±0.4	61.1±0.9	50.7±1.2	70.2±1.1	47.5±1.1	57.7±1.8	80.2±1.7	67.4±1.4
	ZDDA+PD [15]	67.4±4.2	85.7±4.0	87.6±4.7	55.1±3.7	49.2±2.5	59.5±5.9	42.6±3.4	23.7±7.3	75.5±1.7	52.0±2.7
	ZDDA+DC [15]	56.7±1.6	78.0±2.0	76.4±2.3	52.4±2.1	45.2±1.9	53.7±1.8	35.8±1.7	20.6±2.1	69.2±2.6	47.6±1.6
	Ours	76.2±1.8	94.6±0.9	95.2±0.6	67.2±1.2	54.3±1.0	73.6±0.8	53.8±1.4	61.4±1.1	84.3±1.2	71.5±1.0

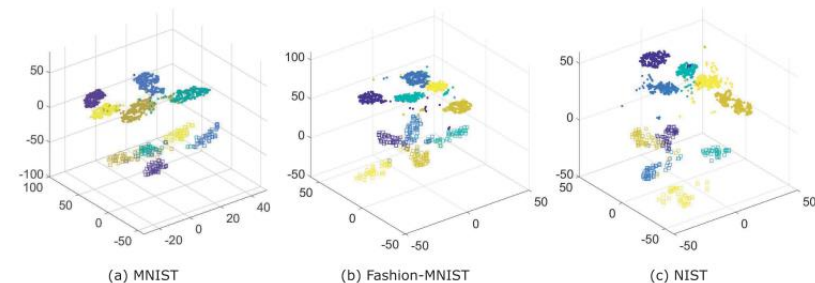


Fig. 6. t-SNE visualization of the representations in two domains: “•” for G-dom and “□” for C-dom (best viewed in color).

- 실험 결과 대부분 좋은 모습을 보임 & 도메인 특성에 맞게 임베딩이 잘 되는 모습을 볼 수 있음

Experiment

TABLE III
AVERAGE ACCURACY (%) ON *Office-Home*

Target	<i>Ar</i>			<i>Cl</i>			<i>Pr</i>			<i>Rw</i>		
Source	<i>Cl</i>	<i>Pr</i>	<i>Rw</i>	<i>Ar</i>	<i>Pr</i>	<i>Rw</i>	<i>Ar</i>	<i>Cl</i>	<i>Rw</i>	<i>Ar</i>	<i>Cl</i>	<i>Pr</i>
DIVA [23]		68.4			61.4			69.7			74.1	
JiGen [24]		72.1			58.5			76.5			78.1	
MATE [22]		65.7			50.5			74.6			65.0	
Epi-FCR [25]		64.0			62.6			76.8			72.8	
MMD-AAE [26]		71.5			57.3			65.7			68.5	
CoCoGAN [16]	66.7	57.6	69.2	61.2	53.4	51.3	69.5	74.0	65.8	74.5	66.4	71.7
ZDDA+DC [15]	67.4	60.9	68.1	53.2	40.6	43.4	61.4	57.0	50.3	68.8	68.4	62.4
Ours	72.6	74.5	73.4	62.3	61.8	66.7	73.8	75.9	82.7	77.7	76.6	75.2
ours+voting		80.4			78.3			90.8			86.4	

TABLE IV
AVERAGE ACCURACY (%) ON *VisDA2017*

N_α	4	5	6	7	8
DIVA [23]	69.8±2.5	68.3±1.6	63.7±3.3	59.0±3.8	57.0±4.5
JiGen [24]	73.2±2.7	68.0±5.2	65.9±3.8	63.7±5.6	59.7±2.6
MATE [22]	68.7±4.0	68.7±3.2	61.4±3.1	56.5±5.6	50.9±4.3
Epi-FCR [25]	72.8±3.4	69.3±5.5	66.9±5.3	61.2±4.9	54.9±5.0
MMD-AAE [26]	68.5±5.8	60.4±2.5	59.4±3.3	53.4±2.4	45.0±2.7
CoCoGAN [16]	44.3±1.3	42.1±0.9	38.3±1.0	37.9±1.2	36.3±1.4
ZDDA+DC [15]	52.3±1.8	47.5±1.6	41.7±1.2	40.8±0.9	36.6±1.0
Ours	76.8±1.2	74.3±1.4	70.4±0.9	68.4±1.7	64.4±1.2

- 두 데이터 셋은 G-dom to C-dom으로 고정
- 다른 방법들과 비교해서 좋은 성능을 보임

Experiment

- 각 스텝마다 사용했던 방법들을 제거해서 중요도를 알아보기 위한 실험
- 다들 각자 훌륭한 역할을 수행했지만 IRT GAN 모델에서 Non shared weigh를 초기 weigh로 설정한 방식이 가장 성능에 큰 역할을 끼치지 못했음
- 이에 반해 GANShift를 통해 도메인 차이에 대한 정보를 주는 것과 IRT CoGAN에서 학습한 도메인간의 차이를 element wise 차 계산으로 하는 것 순으로 중요도가 나왔음
- 밑에 그래프는 옵티마이저 및 배치 크기에 따라 성능이 어떻게 바뀌는지 하이퍼 파라미터 튜닝 과정
- 기존 연구처럼 Adam이 무난하게 가장 좋은 결과를 보임

TABLE V

ABLATION STUDY ON *Office-Home*: “✓” INDICATES THAT THE COMPONENT OF OUR METHOD IS ADOPTED AND “—” INDICATES THAT THE ALTERNATIVE IS ADOPTED

X_{fer}	Shift	Init	$Pr \rightarrow Ar$	$Rw \rightarrow Cl$	$Rw \rightarrow Pr$	$Ar \rightarrow Rw$	Avg.
—	—	—	48.9 ± 3.7	40.3 ± 2.8	46.0 ± 2.3	45.6 ± 1.2	45.20
—	—	✓	55.6 ± 2.9	50.7 ± 3.5	52.5 ± 3.5	56.3 ± 1.1	53.78
—	✓	—	52.1 ± 3.8	53.0 ± 2.1	59.3 ± 1.3	56.5 ± 2.5	55.23
✓	—	—	54.4 ± 2.9	49.5 ± 1.8	63.3 ± 3.4	57.2 ± 3.0	56.10
—	✓	✓	67.6 ± 1.2	63.4 ± 2.9	73.9 ± 2.7	65.1 ± 2.9	67.50
✓	—	✓	71.4 ± 2.4	58.0 ± 2.6	69.1 ± 3.7	70.5 ± 2.7	67.25
✓	✓	—	70.0 ± 1.2	60.9 ± 2.5	76.4 ± 2.3	71.7 ± 2.1	69.75
✓	✓	✓	74.5 ± 2.5	66.7 ± 1.3	82.7 ± 2.2	77.7 ± 3.8	75.40

