Domain Intersection and Domain Difference

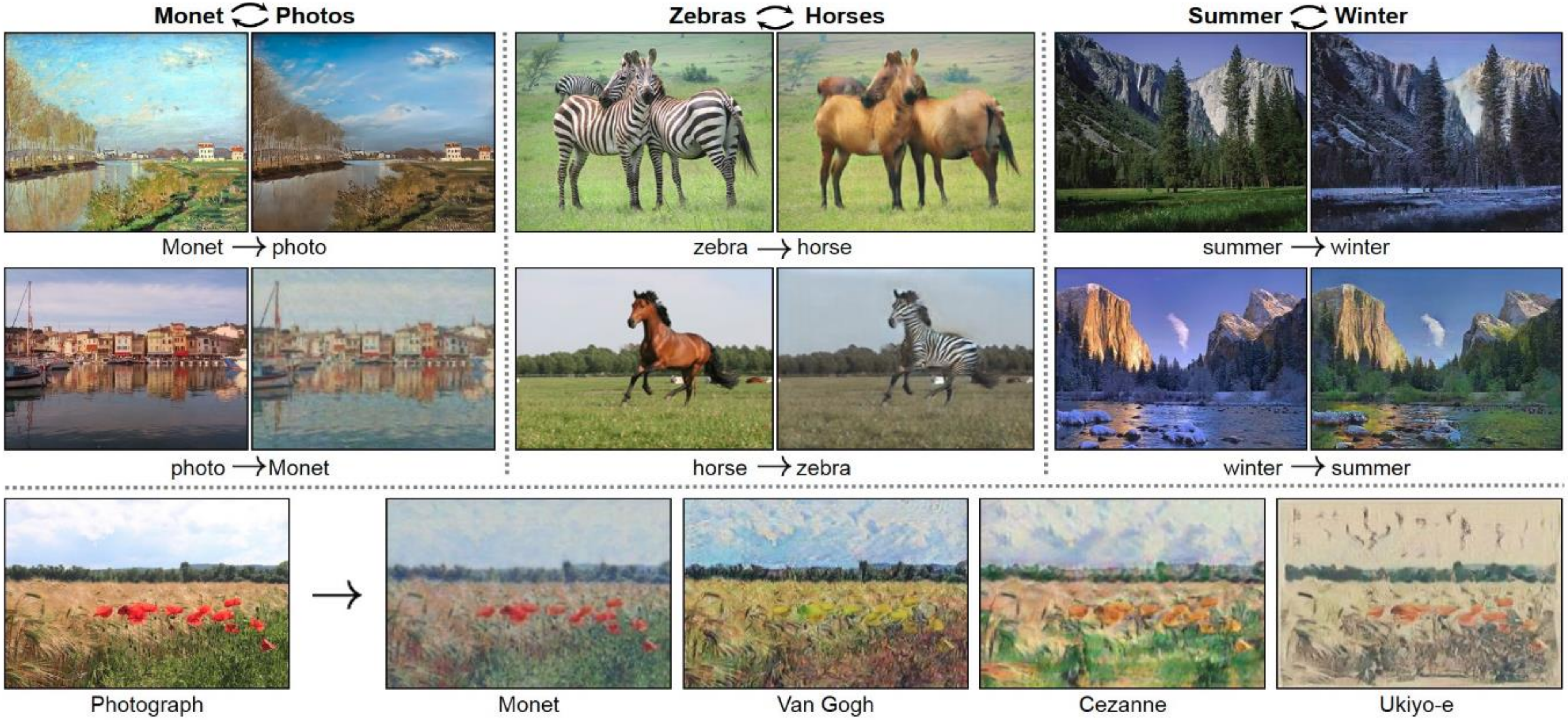
Sagie Benaim[1], Michael Khaitov[1], Tomer Galanti[1], Lior Wolf[1,2]

[1]Tel Aviv University

[2]Facebook AI Research

TEL AVIV UNIVERSITY

Image to Image Translation



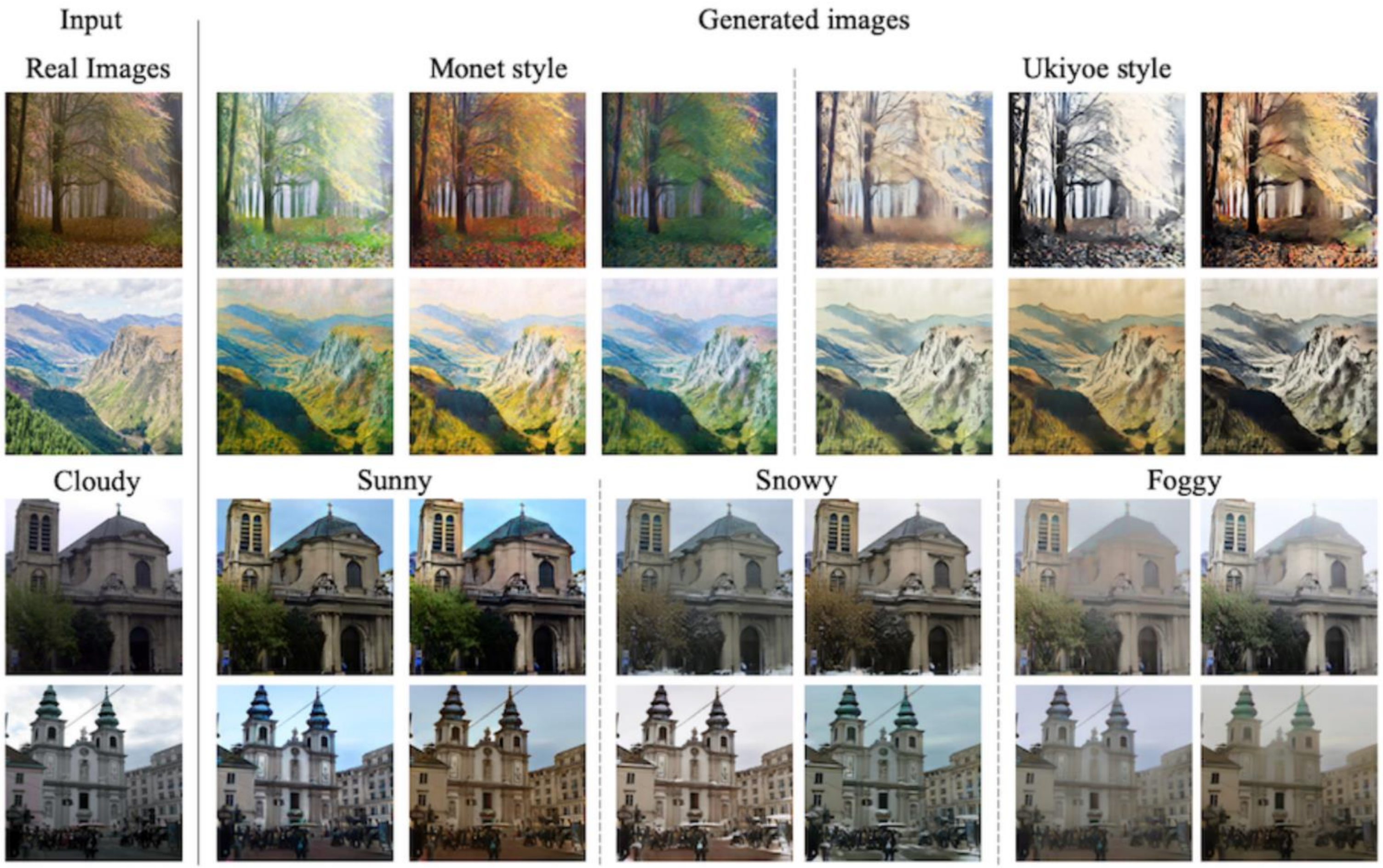
MUNIT: Style and Texture Changes

Sketch to Image Translation

(a) edges $\leftrightarrow$ shoes (b) edges $\leftrightarrow$ handbags

Huang et al., ECCV 2018

DRIT, DRIT++: Similar Textural and Style Changes



Lee et al., ECCV 2018

Cannot Transfer Content!

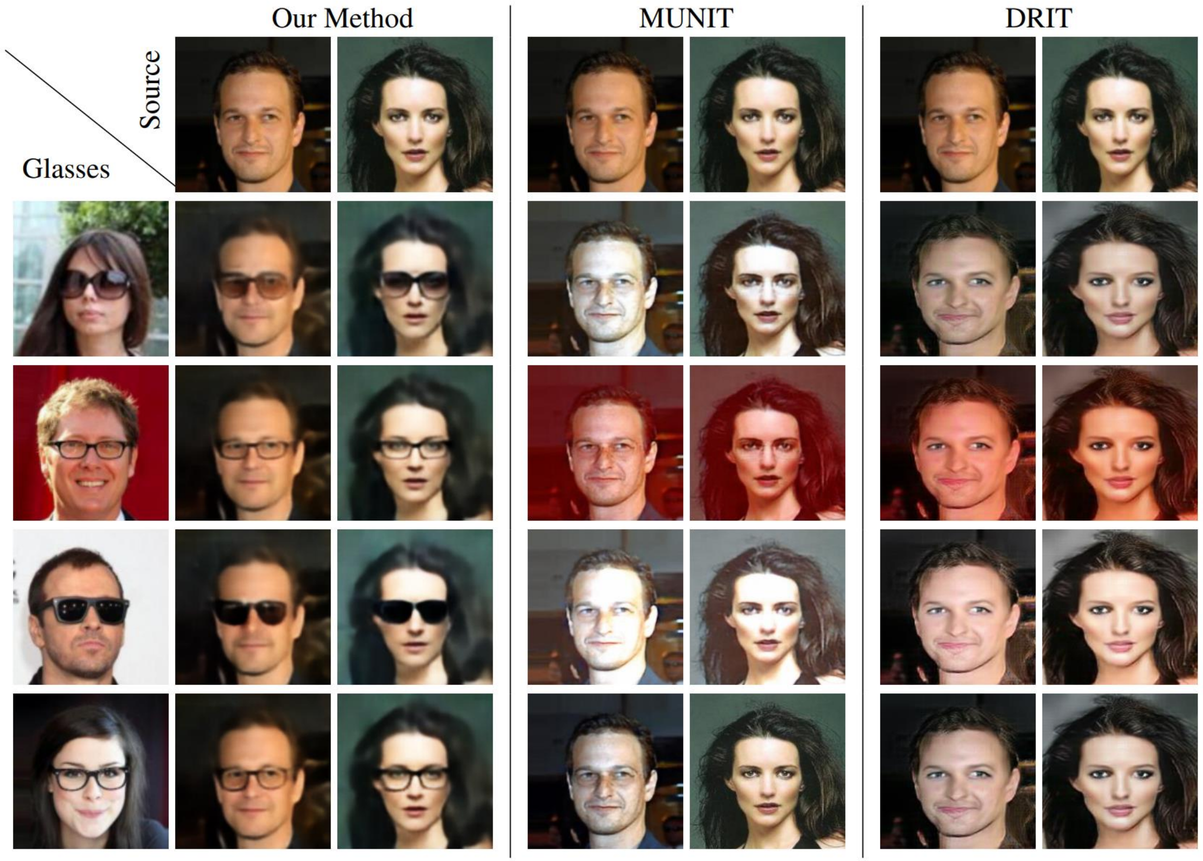


Figure 2: Glasses transfer. Our method vs literature baselines. Each image combines the domain A image in the top row, with the content of the guide image on the left column.

Press et al, ICLR 2019

Attribute Transfer

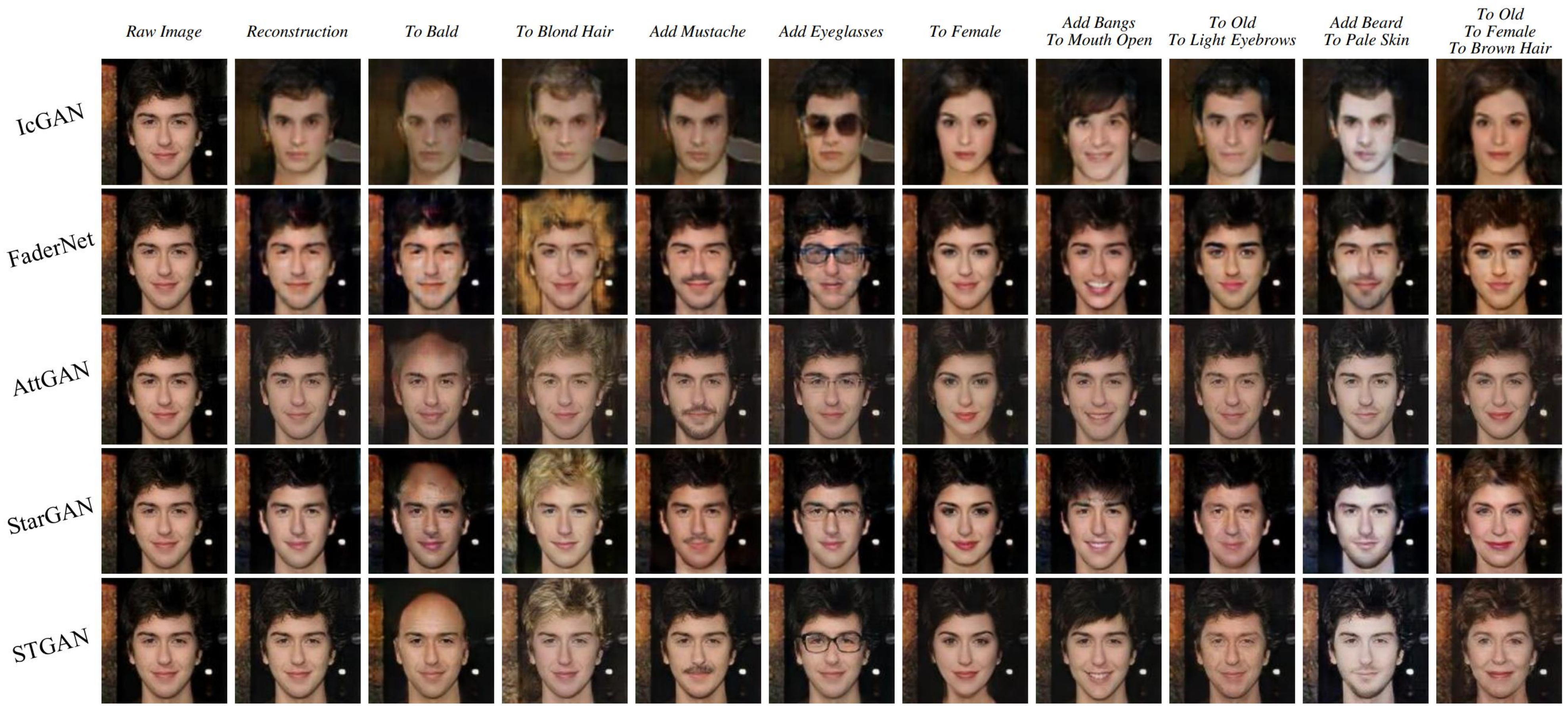


Figure 6: Facial attribute editing results on the CelebA dataset. The rows from top to down are results of IcGAN [26], FaderNet [17], AttGAN [11], StarGAN [7] and STGAN.

Liu et al, CVPR 2019

Only a single Attribute!
For example, Fader Networks:

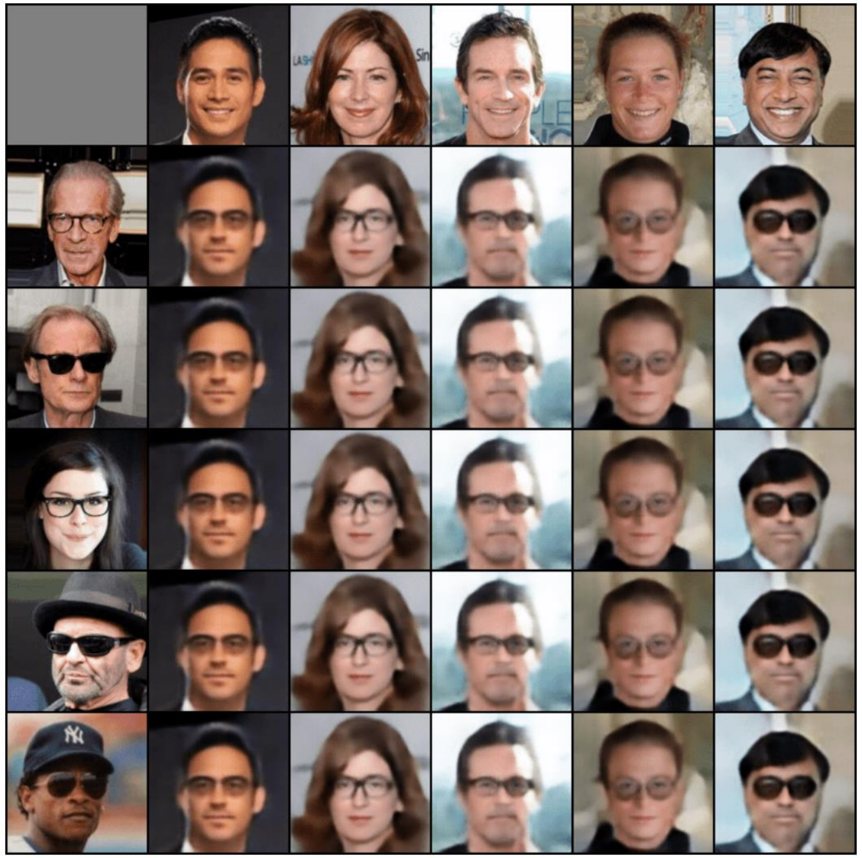


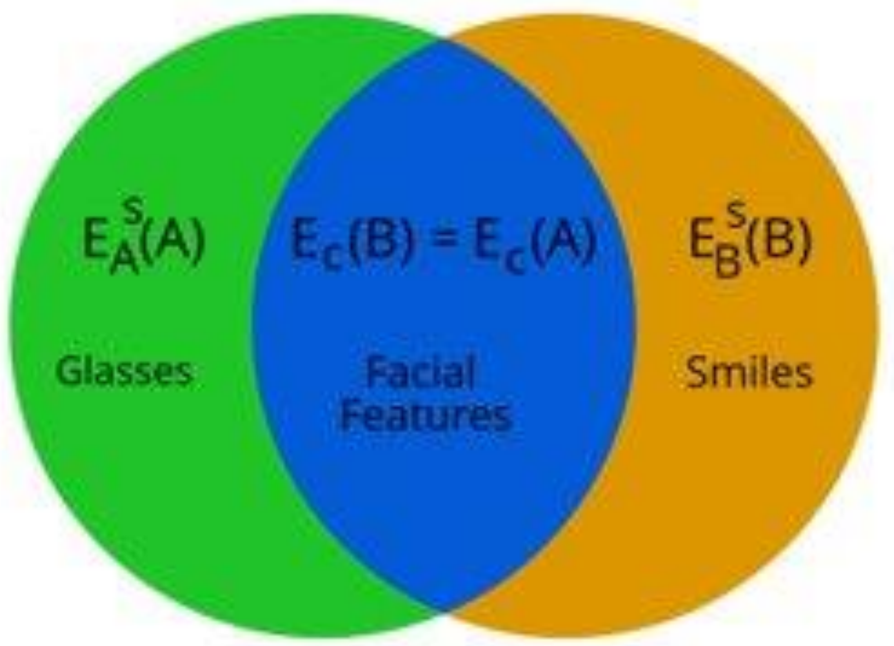
Figure 19. Translation from the domain of smiling persons to the domain of persons with glasses, using the Fader Networks method.

Domain Intersection and Domain Difference

Given two visual domains, disentagle the **separate (domain specific)** information and common **(domain invariant)** information.

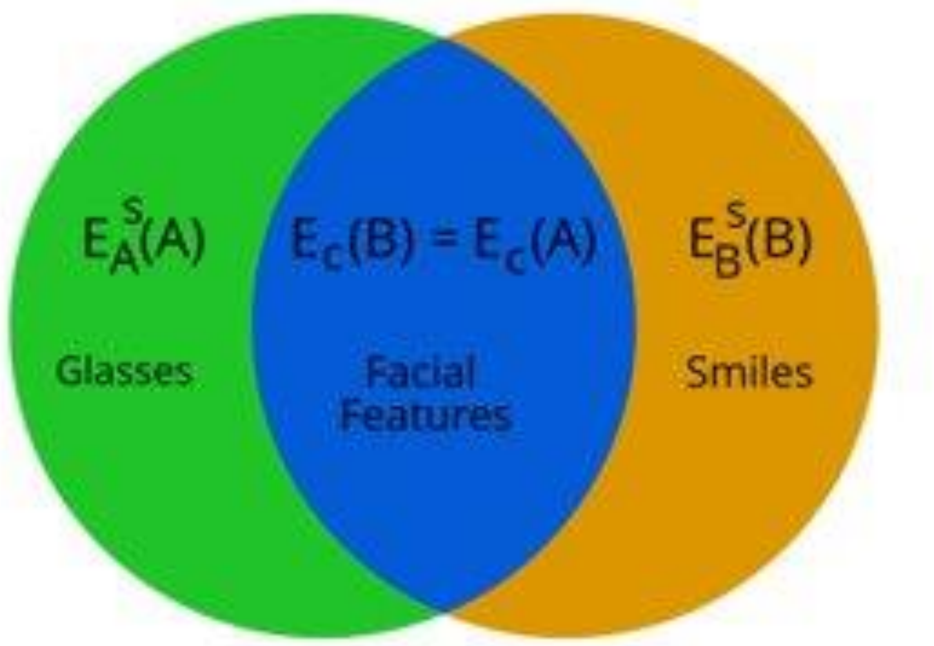
If A is **persons with glasses** and B is **smiling persons,** our method produces three latent spaces:

1. "Common" latent space, $E_c(A) = E_c(B)$. The space of **common facial features.** For $c \in A \cup B, E_c(c)$ is the **facial features of c**.



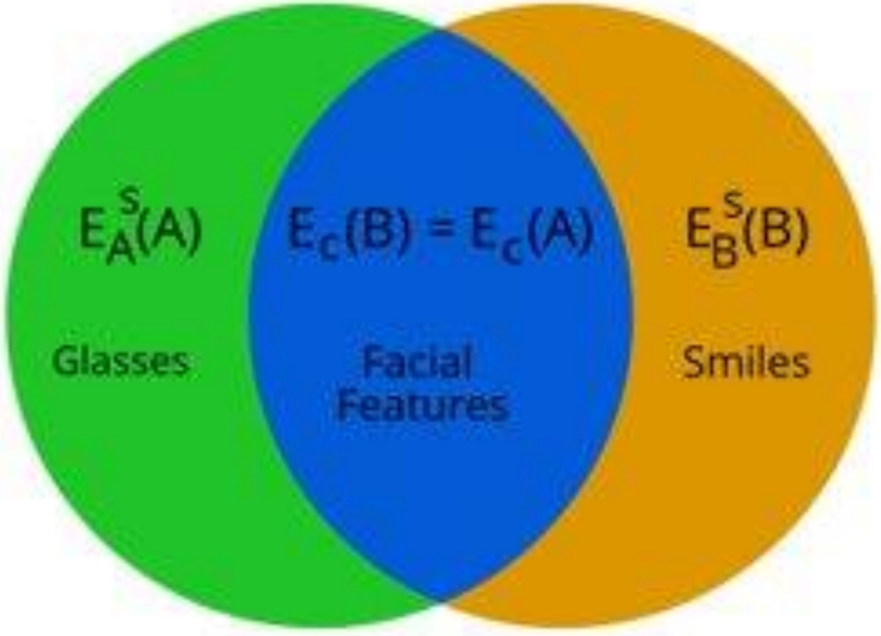
If A is **persons with glasses** and B is **smiling persons,** our method produces three latent spaces:

1. "Common" latent space, $E_c(A) = E_c(B)$. The space of **common facial features.** For $c \in A \cup B, E_c(c)$ is the **facial features of c**.
2. "Separate" latent space for domain A, $E_A^S(A)$. The **space of glasses.** $E_A^S(a)$ is the **glasses of $\boldsymbol{a}$**.

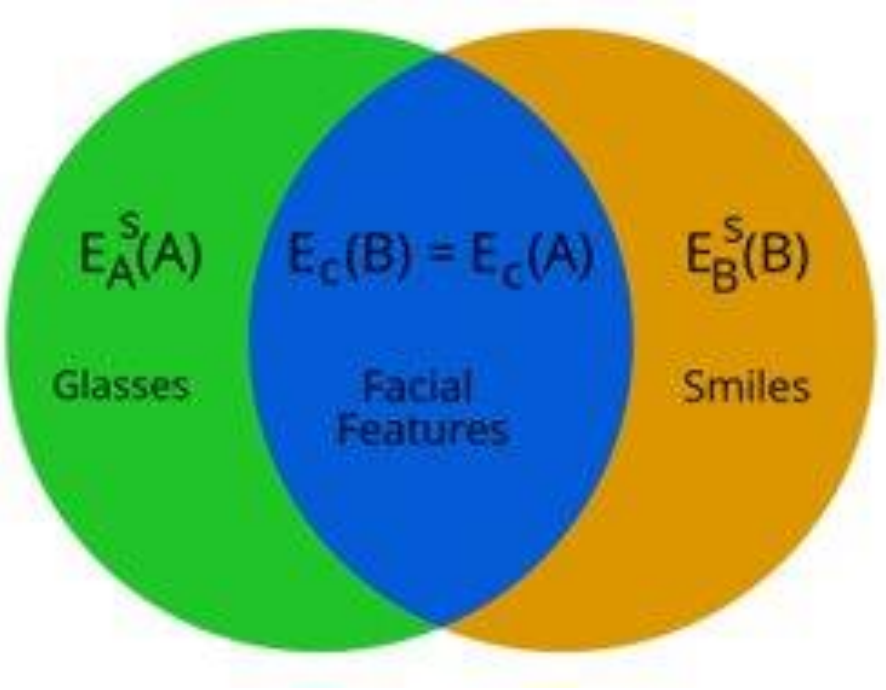


If A is **persons with glasses** and B is **smiling persons,** our method produces three latent spaces:

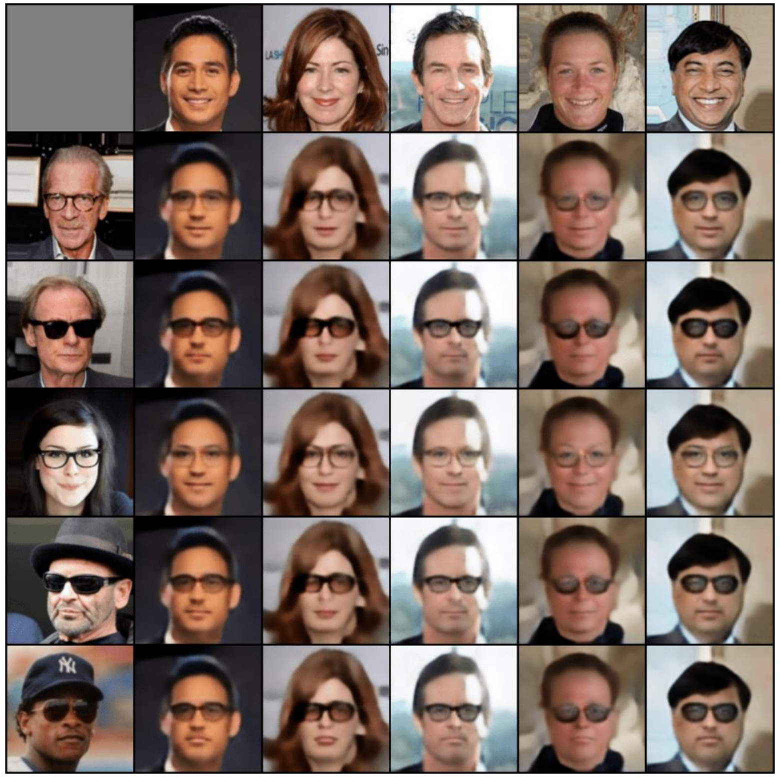
1. "Common" latent space, $E_c(A) = E_c(B)$. The space of **common facial features.** For $c \in A \cup B, E_c(c)$ is the **facial features of c**.
2. "Separate" latent space for domain A, $E_A^s(A)$. The **space of glasses.** $E_A^s(a)$ is the **glasses of $\boldsymbol{a}$**.
3. "Separate" latent space for domain B, $E_B^s(B)$. The **space of smiles.** $E_B^s(b)$ is the **smile of $\boldsymbol{b}$.**



Given this disentangled representation, we generate a visual sample $G(E_c(c), E_A^S(a), E_B^S(b))$, having the **facial features of c, glasses of a, smile of b.**



Smile to Glasses



The "common" (or shared) Loss

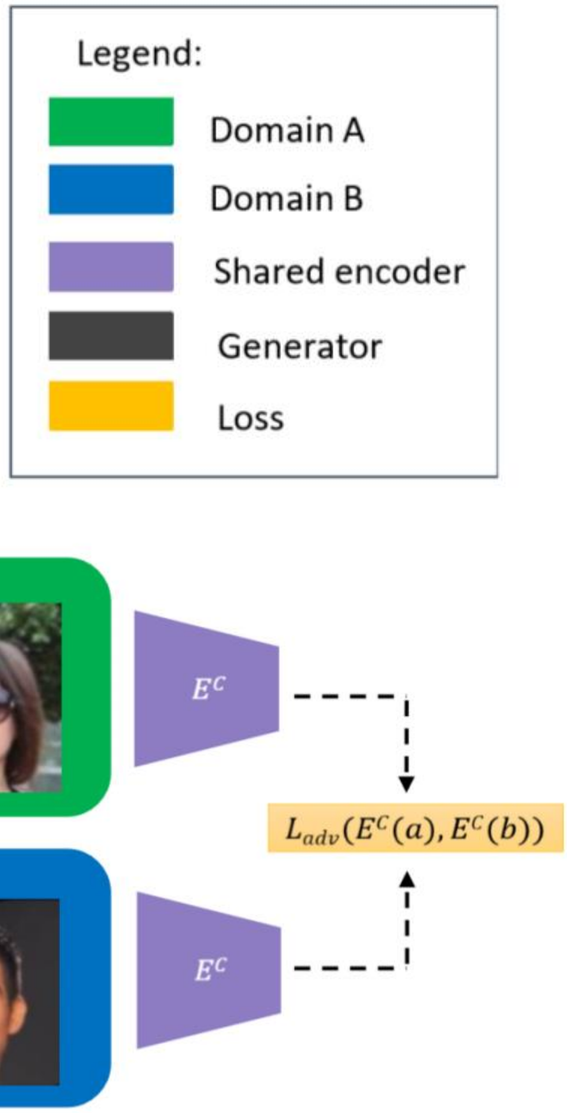
Ensures E_c encodes information common to both domains

Encoder E_c attempts to match distributions of $E_c(A)$ and $E_c(B)$:

$$\frac{1}{m_1}\sum_{i=1}^{m_1} l(d(E^c(a_i)),1) + \frac{1}{m_2}\sum_{j=1}^{m_2} l(d(E^c(b_j)),1)$$

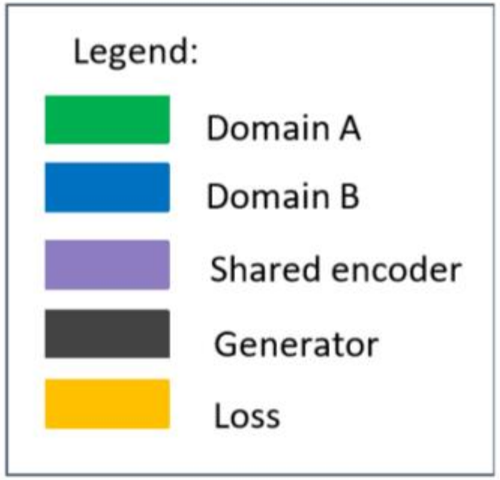
Discriminator d attempts to separate distributions:

$$\mathcal{L}_d := \frac{1}{m_1}\sum_{i=1}^{m_1} l(d(E^c(a_i)),0) + \frac{1}{m_2}\sum_{j=1}^{m_2} l(d(E^c(b_j)),1)$$



(c)

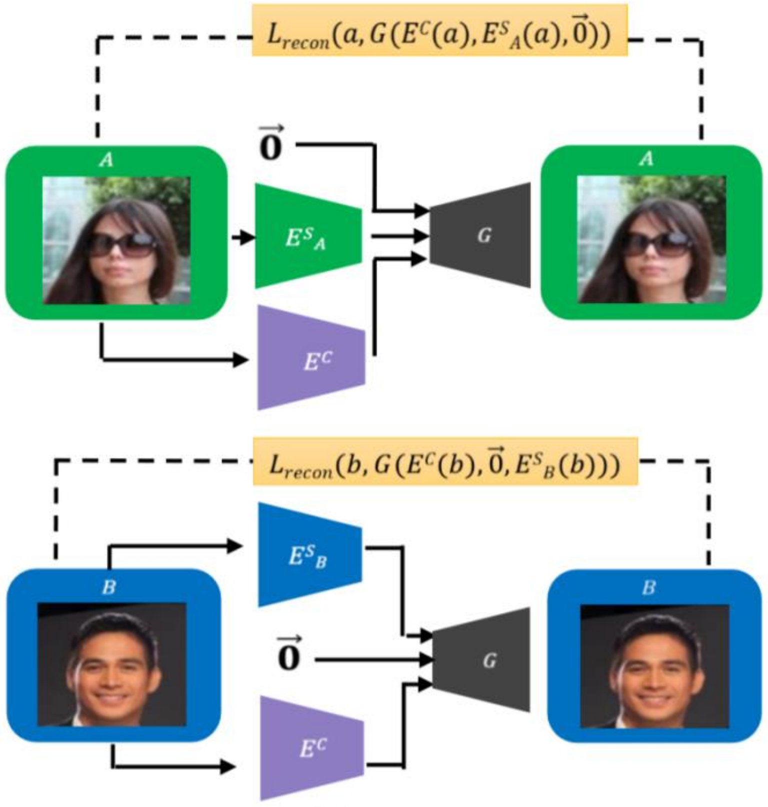
Reconstruction Losses



Ensures the "common" and "separate" encodings contain all the information in A or B

$$\mathcal{L}^A_{recon} := \frac{1}{m_1}\sum_{i=1}^{m_1}\|G(E^c(a_i), E^s_A(a_i), 0) - a_i\|_1$$

$$\mathcal{L}^B_{recon} := \frac{1}{m_2}\sum_{j=1}^{m_2}\|G(E^c(b_i), 0, E^s_B(b_j)) - b_j\|_1$$



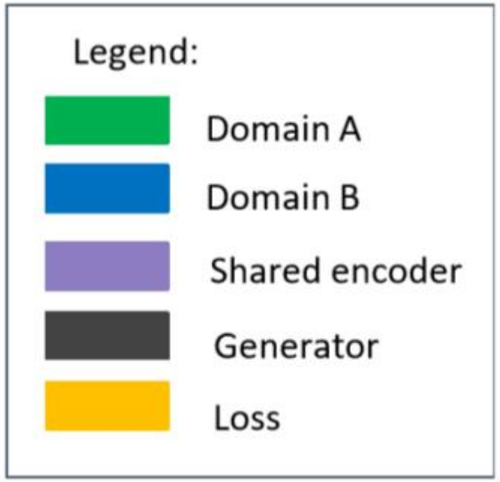
(b)

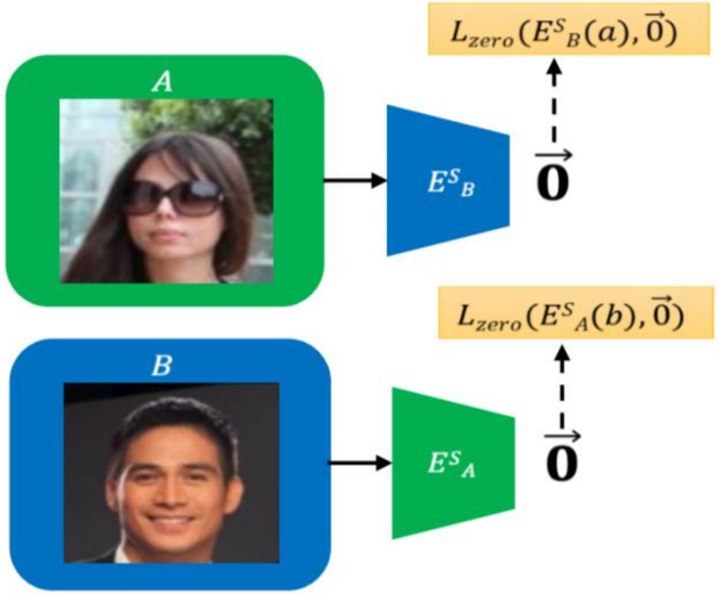
"Zero" Loss

Ensures the separate encoder of A (resp. B) does not encode information about B (resp. A)

$$\mathcal{L}^A_{zero} := \frac{1}{m_2}\sum_{j=1}^{m_2} \|E^s_A(b_j)\|_1$$

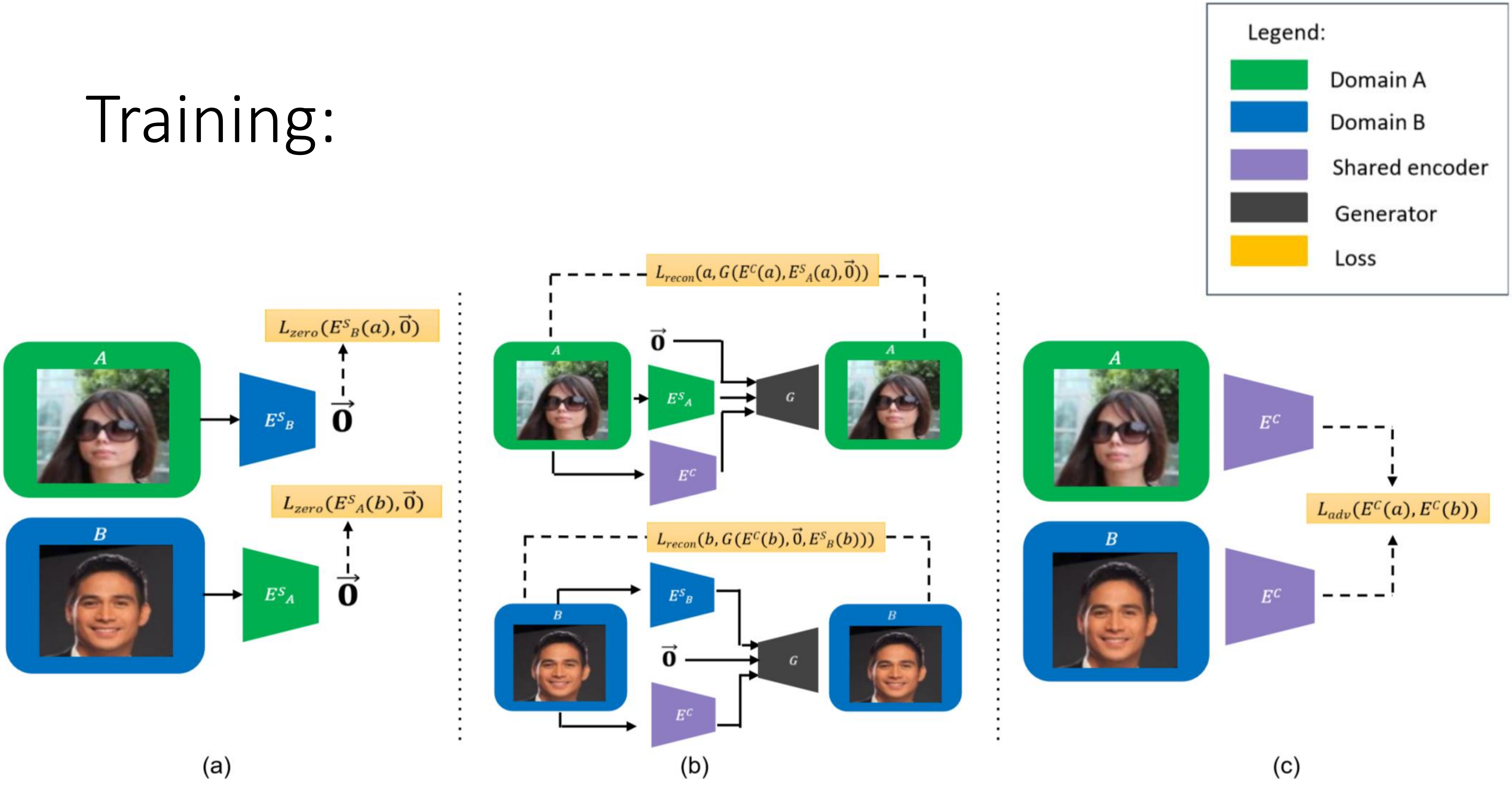
$$\mathcal{L}^B_{zero} := \frac{1}{m_1}\sum_{i=1}^{m_1} \|E^s_B(a_i)\|_1$$





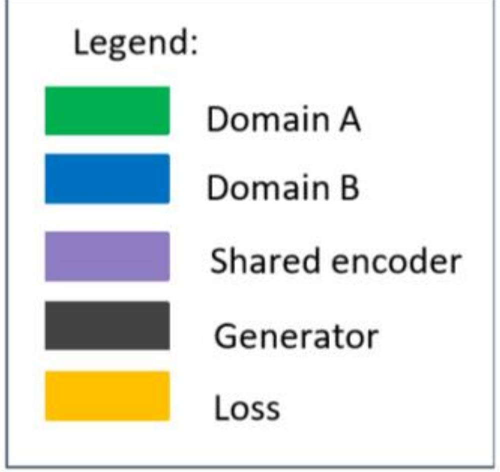
(a)

Training:



Legend:
Domain A
Domain B
Shared encoder
Generator
Loss
A
B
$E^S{}_B$
$E^S{}_A$
$\vec{0}$
$L_{zero}(E^S{}_B(a), \vec{0})$
$L_{zero}(E^S{}_A(b), \vec{0})$
$L_{recon}(a, G(E^C(a), E^S{}_A(a), \vec{0}))$
$L_{recon}(b, G(E^C(b), \vec{0}, E^S{}_B(b)))$
E^C
G
$L_{adv}(E^C(a), E^C(b))$
(a)
(b)
(c)

Inference:

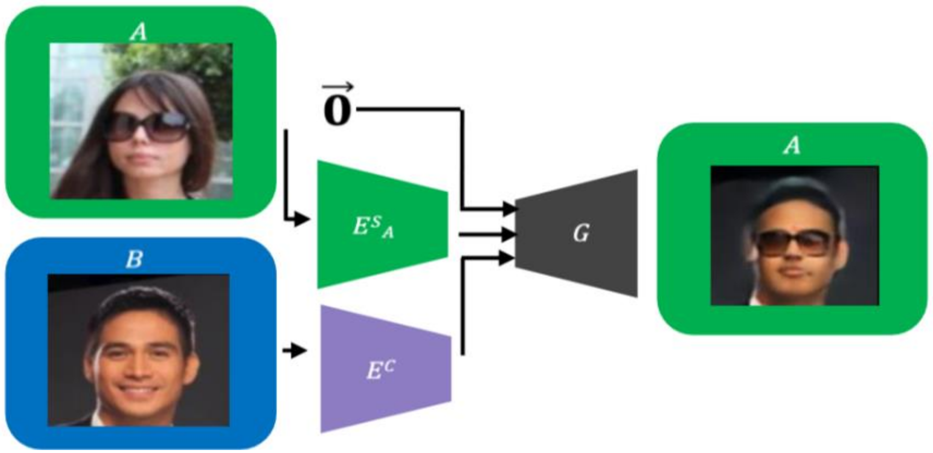


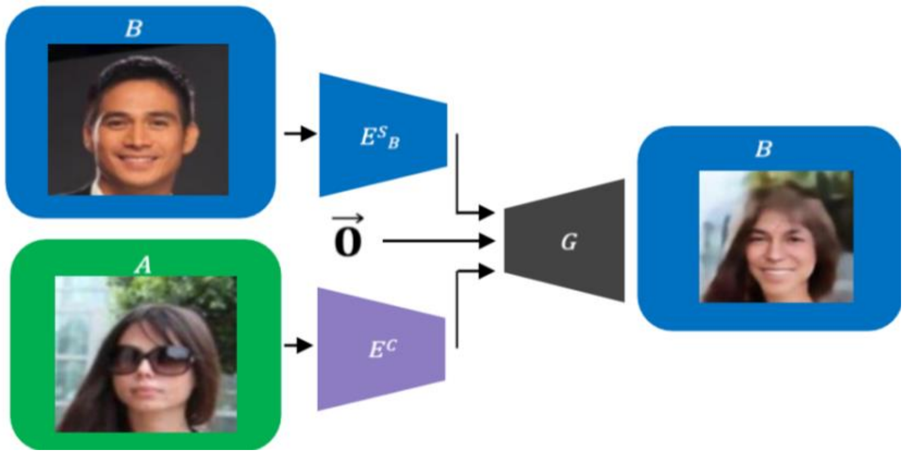
$$G(E_c(b), E_A^S(a), 0)$$

remove b's smile
add a's glasses

$$G(E_c(a), 0, E_A^S(b))$$

remove a's glasses
add b's smile





Results

Beard to Smile

Glasses to Smile

Glasses ∩ Smile

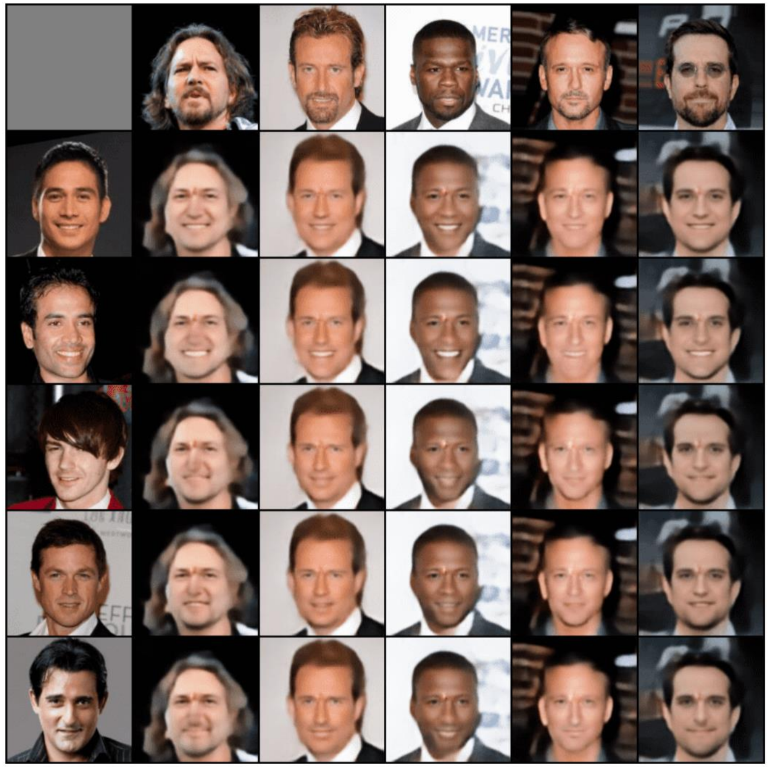


Figure 8. Translating from the domain of persons with facial hai
to the domain of smiling persons.

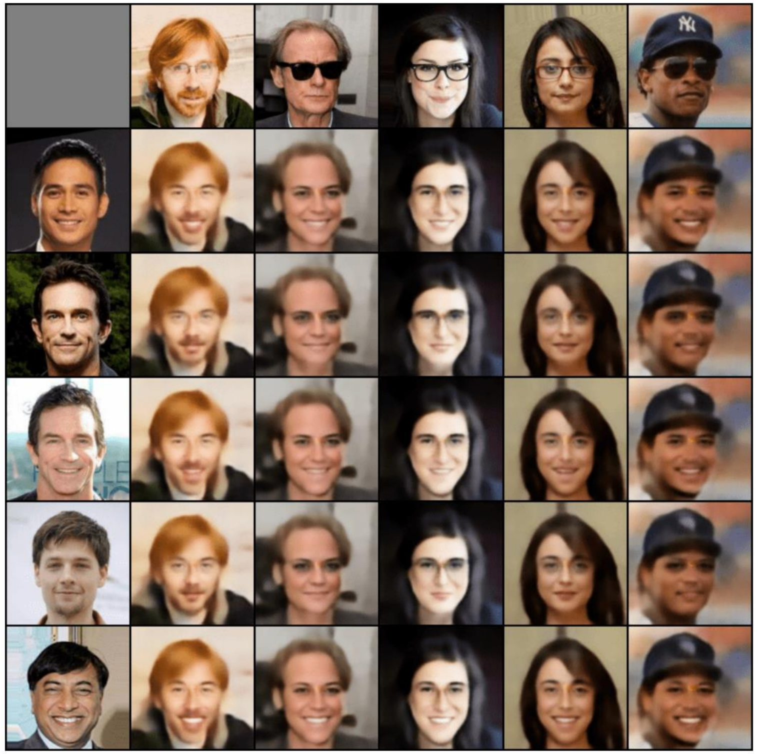
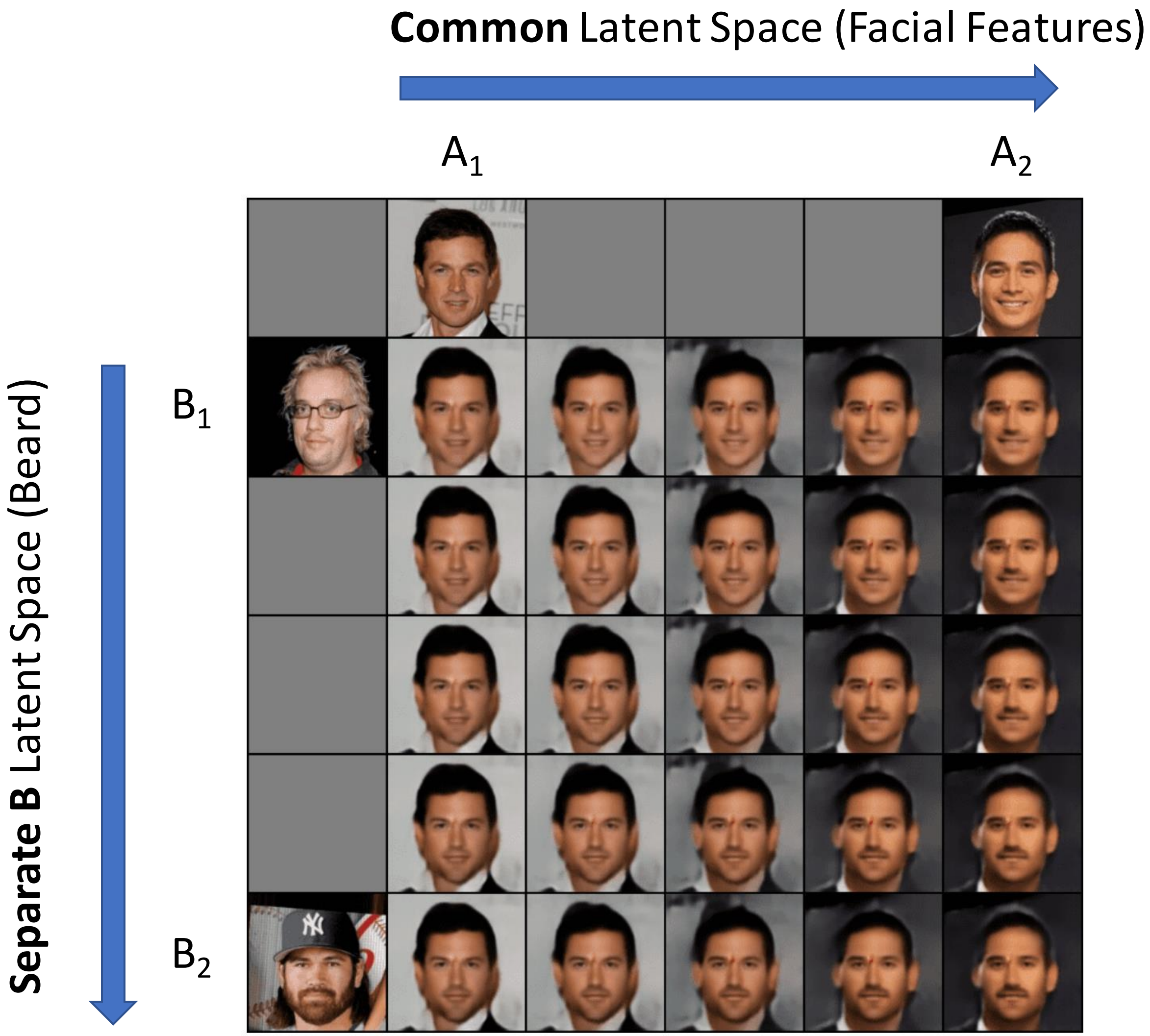
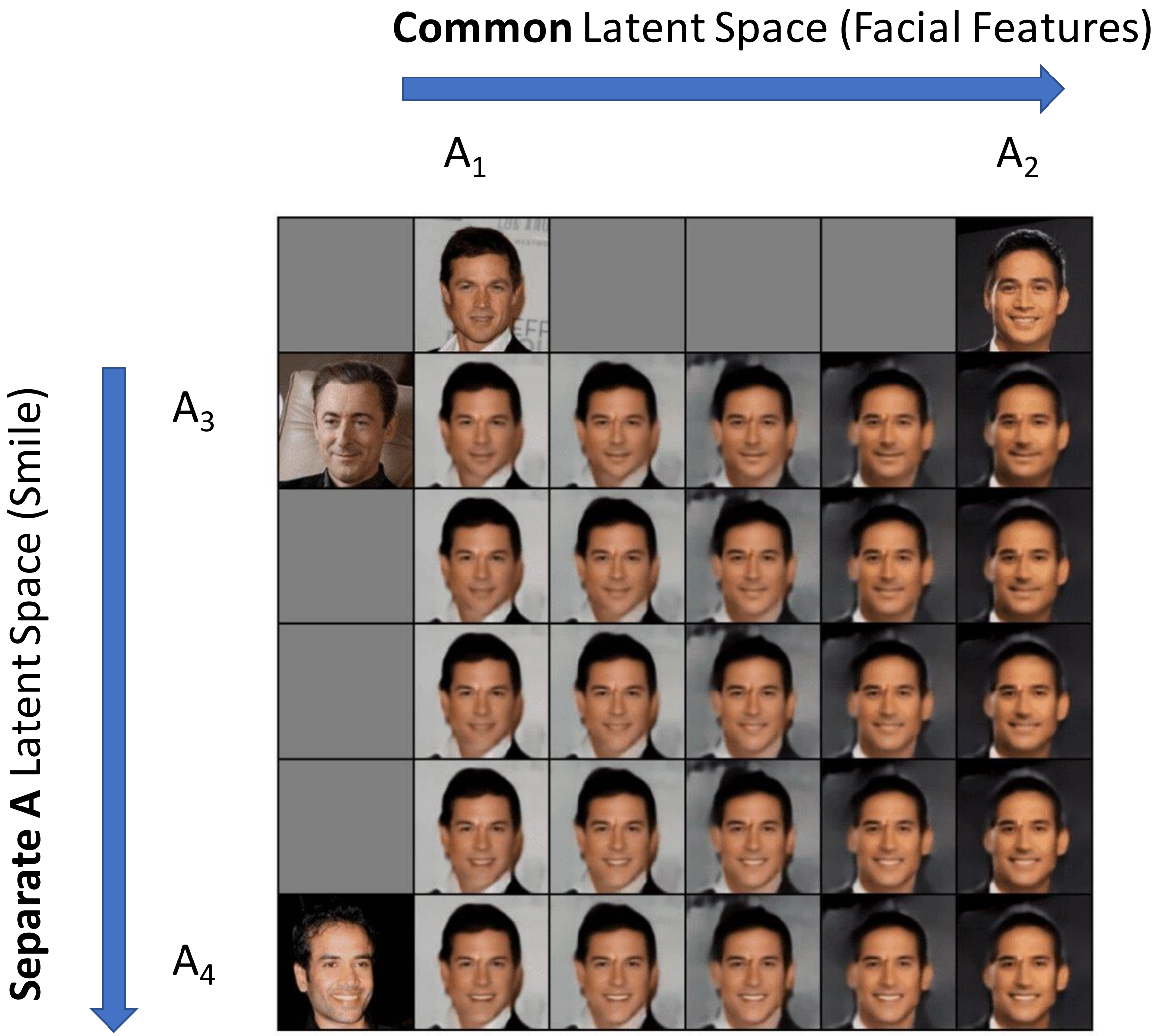


Figure 7. Translating from the domain of persons with glasses to the domain of smiling persons (reverse translation to Fig. 2 in main report)

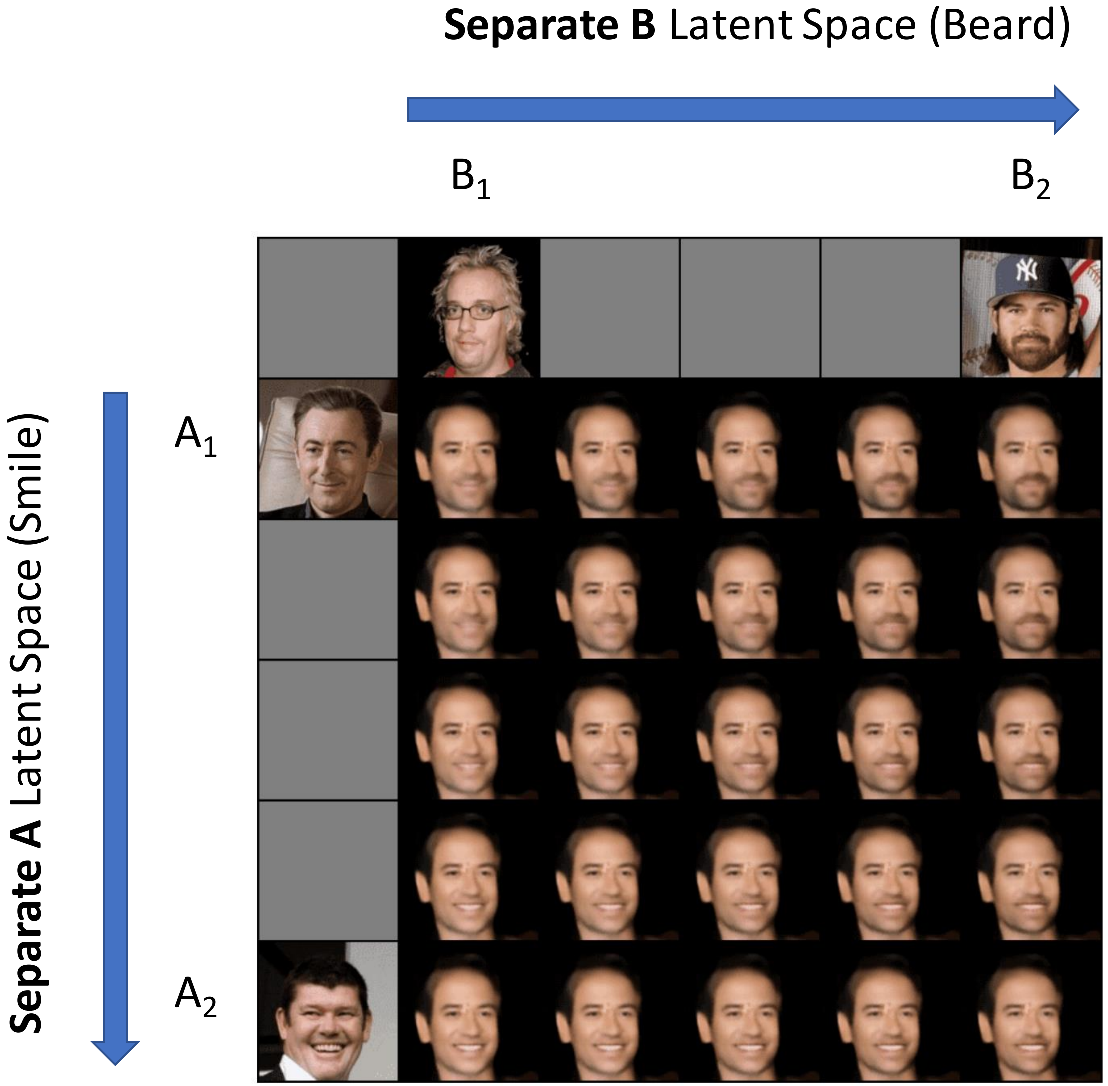
Interpolations



Interpolations



Interpolations



Numerical Results: Pretrained Classifier

	Smile To Glasses	Glasses To Smile	Facial Hair To Smile	Smile To Facial Hair	Facial Hair To Glasses	Glasses To Facial Hair
Fader networks [15]	76.8%	97.3%	95.4%	84.2%	77.8 %	85.2%
Guided content transfer [20]	45.8%	92.7%	85.6%	85.1%	38.6%	82.2%
MUNIT [12]	7.3%	9.2%	9.3%	8.4%	7.3%	8.5%
DRIT [16]	8.5%	6.3%	6.3%	10.3%	8.6%	10.1%
Ours	91.8%	99.3%	93.7%	87.1%	93.1%	97.2%

Table 1. We pretrain a classifier to distinguish between samples in A (e.g. images of persons with glasses) and samples in B (e.g. images of persons with smile). We then sample $a \in A$, $b \in B$ from the test samples and check the membership of the generated image $G(E^c(b), E^s_A(a), 0))$ in A. Similarly, in the reverse direction, we check the membership of $G(E^c(a), 0, E^s_B(b))$ in B.

Numerical Results: User Study

- Q1: Is the specific attribute of A (e.g smile) removed?
- Q2: Is the guided image b specific attribute (e.g glasses) added?
- Q3: Is the identify of a's image preserved?

	Smile To Glasses	Glasses To Smile	Facial Hair To Smile	Smile To Facial Hair	Facial Hair To Glasses	Glasses To Facial Hair
Question (1) ours	4.74 $\pm$0.13	4.30 $\pm$0.21	4.26 $\pm$0.20	4.30 $\pm$0.15	4.18 $\pm$0.17	4.50 $\pm$0.18
Question (2) ours	3.92 $\pm$0.16	4.45 $\pm$0.12	4.03 $\pm$0.15	3.34 $\pm$0.17	3.85 $\pm$0.20	3.95 $\pm$0.22
Question (3) ours	3.95 $\pm$0.23	3.20 $\pm$0.24	3.24 $\pm$0.25	3.22 $\pm$0.27	3.49 $\pm$0.22	3.39 $\pm$0.23
Question (1) for [20]	3.67 $\pm$0.17	4.16 $\pm$0.18	3.39 $\pm$0.19	3.34 $\pm$0.13	4.24 $\pm$0.12	3.15 $\pm$0.15
Question (2) for [20]	1.87 $\pm$0.35	4.42 $\pm$0.22	3.00 $\pm$0.32	2.67 $\pm$0.33	2.20 $\pm$0.42	3.30 $\pm$0.22
Question (3) for [20]	3.95 $\pm$0.15	2.93 $\pm$0.22	3.37 $\pm$0.25	3.40 $\pm$0.27	3.43 $\pm$0.28	3.75 $\pm$0.20

Table 2. Given 20 randomly selected images $a \in A$ and $b \in B$, we consider the generated image $G(E^c(a), 0, E^s_B(b)))$ and ask if (1) a's separate part is removed (2) b's separate part is added (3) a's common part is preserved (similarly in the reverse direction). Mean opinion scores in the range of 1 to 5 are reported, where higher is better.

Domain Adaptation

- Our disentangled representation is useful for **Unsupervised** Domain Adaptation: **No labels at all.**
- A pretrained classifier is used to evaluate the percentage of images mapped to the same label in the target domain.
- Given an MNIST digit a, we randomly sample an SVHN digit b and consider the translation to SVHN as $G(E_c(a), 0, E_A^s(b))$.
- Achieve **SOTA:** MNIST to SVHN: 61.0%, Reverse: 41.0%

Theory

- Under mild assumptions (such as our losses being minimized):
  - $E^c(A)$ and $E^s_A(A)$ are independent (Similarly for B).
  - $E^c(A)$ captures the information underlying $e^c(A)$ (Similarly for B).
  - $E^s_A(A)$ holds the information underlying $e^s_A(A)$ (Similarly for B).
  - I.e. our losses are both **necessary and sufficient** for the desired **disentanglement**.

"Masked Based Unsupervised Content Transfer" (ICLR 2020)

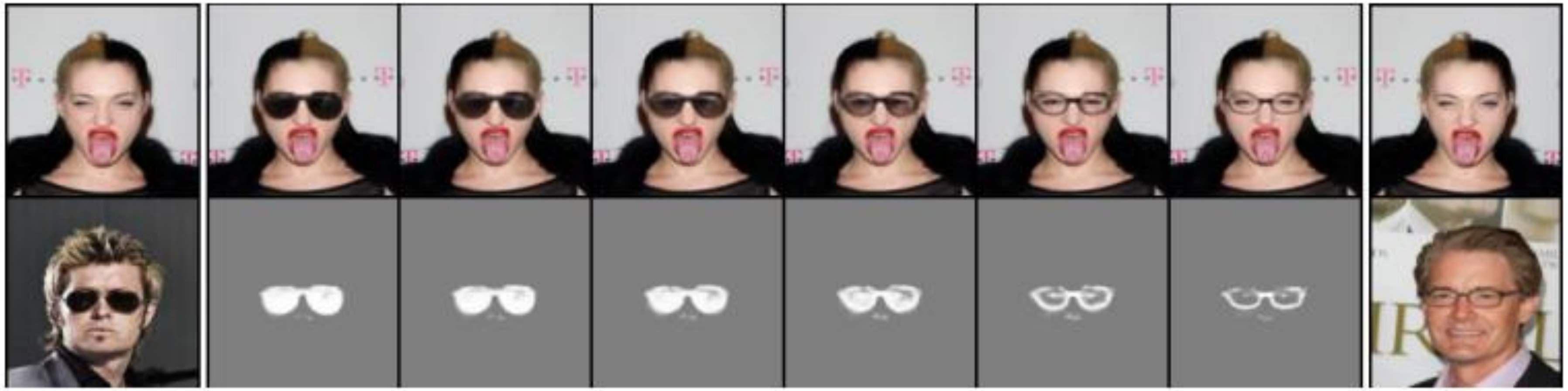
- Only a local change in the target is needed
- Learn a mask and adapt only the area in the masked area

Two Attributes

Smile to Glasses

Additional Content Transfer

Interpolation



Attribute Removal

Figure 6: Attr removal.

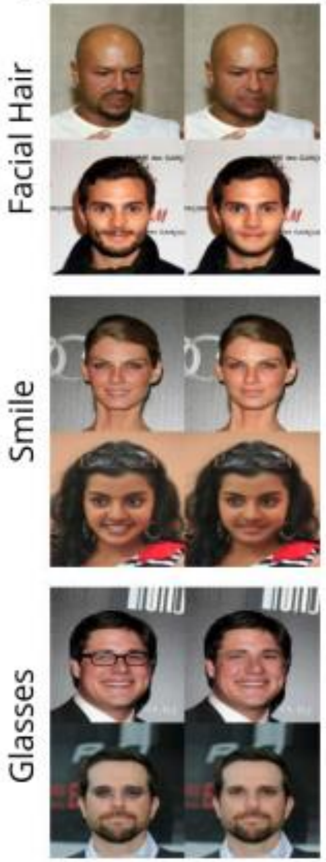


Table 6: Attribute removal for the task of Smile, Facial hair and Glasses.

Task	Method	KID	FID	Class.	Sim.
Smile	Ours	2.6 ± 0.4	120.0 ± 2.6	96.9%	0.96
	Press et al.	15.0 ± 0.6	167.7 ± 0.3	96.9%	0.81
	He et al.	4.1 ± 0.4	127.7 ± 4.5	96.9%	0.95
	Liu et al.	4.3 ± 0.3	129.0 ± 3	98.4%	0.92
	Fader	11.3 ± 0.7	155.6 ± 4.7	93.7 %	0.89
Mustache	Ours	1.9 ± 0.5	119.0 ± 0.8	95.3 %	0.95
	Press et al.	16.6 ± 0.8	175.9 ± 1.4	100.0%	0.80
	He et al.	4.6 ± 0.5	130.0 ± 3.0	87.5%	0.96
	Liu et al.	14.0 ± 0.6	160.0 ± 3.3	87.5%	0.85
	Fader	14.1 ± 0.6	162.6 ± 1.5	98.4 %	0.76
Glasses	Ours	5.2 ± 0.5	136.5 ± 2.6	99.2%	0.87
	Press et al.	15.3 ± 0.5	172.0 ± 4.7	100.0%	0.73
	He et al.	8.3 ± 0.9	141.4 ± 6.8	100.0%	0.84
	Liu et al.	6.8 ± 0.3	141.8 ± 4.8	98.4%	0.86
	Fader	12.5 ± 0.3	137.7 ± 4.2	100.0%	0.76

Out of Domain Manipulation

(a) (b)

Figure 23: Out of domain translation. (a) Results on extremely out of domain images. (b) Results obtained by manipulating LFW images.

Semi Supervised Segmentation Using Class Information

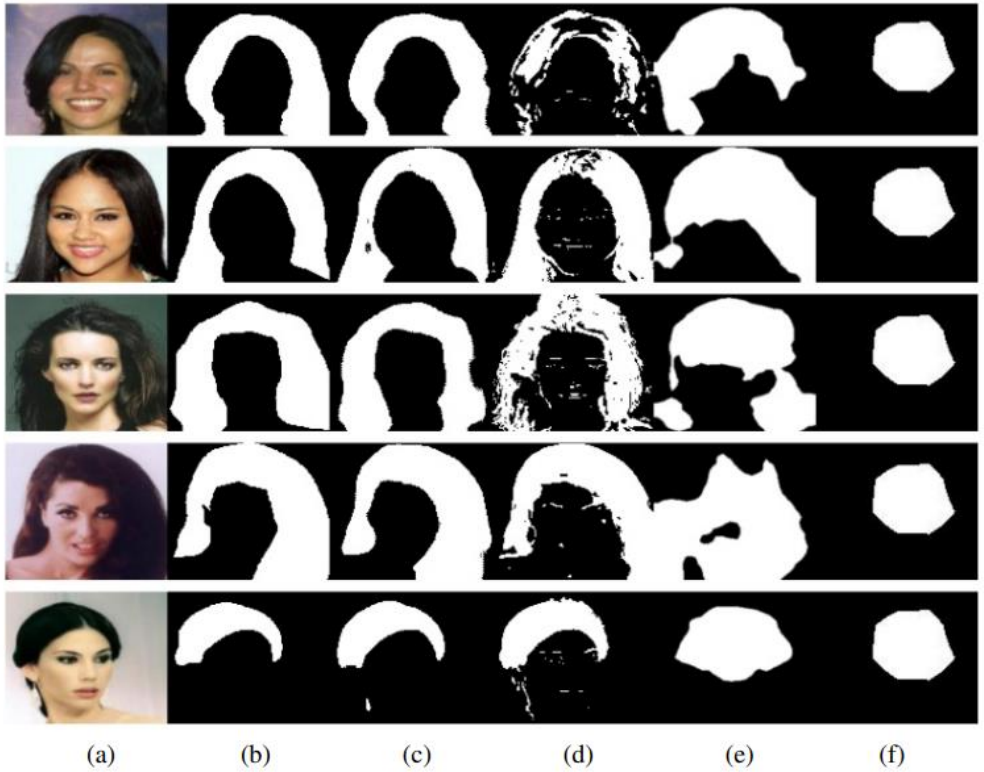
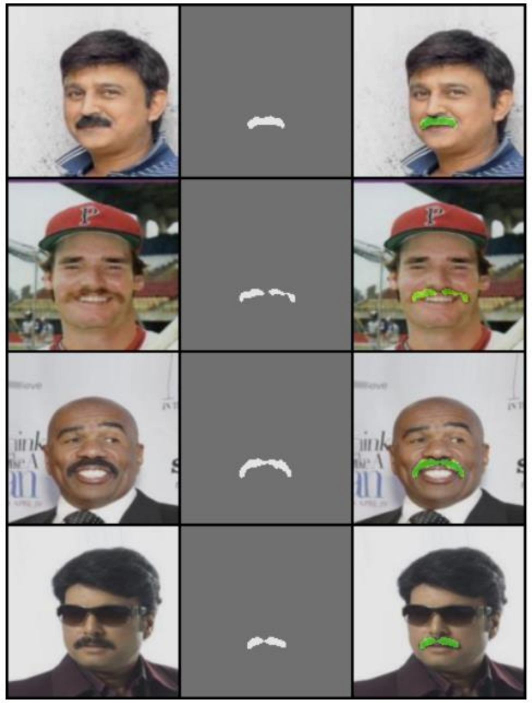


Figure 35: Additional Segmentation results for of women's hair. (a) original image, (b) ground truth segmentation, (c) our results, (d) the results of Press et al. (2019), (e) the results of Ahn & Kwak (2018), (f) results of CAM.



Code and paper available online:
https://github.com/sagiebenaim/DomainIntersectionDifference

Questions?