Domain Intersection and Domain Difference

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

[1]School of Computer Science, Tel Aviv University
[2]Facebook AI Research

Abstract

We present a method for recovering the shared content between two visual domains as well as the content that is unique to each domain. This allows us to map from one domain to the other, in a way in which the content that is specific for the first domain is removed and the content that is specific for the second is imported from any image in the second domain. In addition, our method enables generation of images from the intersection of the two domains as well as their union, despite having no such samples during training. The method is shown analytically to contain all the sufficient and necessary constraints. It also outperforms the literature methods in an extensive set of experiments.

1. Introduction

In unsupervised mapping between visual domains, the algorithm receives two unmatched sets of samples: one from domain A and one from domain B. It then learns a mapping function that generates, for each sample a in domain A, a matching sample in B.

Without a supervision in the form of pairs of matched samples, the problem, like other unsupervised tasks, can be ambiguous [5]. However, it is natural to expect that a pair of samples (a, b), one from each domain, would be considered matching, if there is a significant amount of shared content between a and b. The more content is shared, the stronger the link between the two samples.

Therefore, one can consider the intersection of two visual domains A and B as a domain that contains all of the information that is common to the two domains. This shared domain needs not be visual, and it can contain information that is encoded (latent information).

Turning our attention to the information that complements the shared information, each domain also has a separate, unshared part, which is domain-specific in the context of the two domains.

When mapping a sample a from domain A to B, we can, therefore, consider three types of information. The part of a that is in the shared domain needs to remain fixed under the transformation. The part of a that is specific to domain A is discarded. Lastly, the part of the generated sample in B that is specific to this domain is arbitrary.

While many unsupervised domain mapping methods do not specify the component that is specific to the second domain, some of the recent methods rely on a sample in B to donate this information. Such methods are called guided image to image translation methods. The literature has two types of such methods: those that borrow the style from the image in B, assuming that the domain specific information is a type of visual style [10, 14], and a recent method [18] which assumes that domain A is a subset of domain B, which does not contain any information that is not present in B. In both cases, these assumptions seem too strong.

Our method is able to deal with the two separate domains in a symmetric way, without assuming that domain B can contribute only a different style and without assuming that A is a degenerate subset of B. The method employs a set of loss terms that lead, as our analysis shows, to a disentanglement between the three types of information that exist in the two domains.

As a result, our method enables a level of control that is unprecedented in mapping image across domains. It allows us to take the specific part that belongs to domain A from one image, the specific part of domain B from another image, and the shared part from either image or from a third image. In addition, each of the three parts can be interpolated between different samples, and the domain specific parts can be eliminated altogether.

1.1. Previous Work

In image to image translation, the algorithm is provided with two independent datasets from two different domains. The goal is to learn a transformation of samples from the first domain to samples from the second domain. These transformations are often implemented by a deep neural network that has an encoder-decoder architecture.

The early solutions to this problem assumed the existence of an invertible mapping y from the first domain to the second domain. This mapping takes a sample a in domain A and maps it to an analog sample in domain B. The cir-

cularity based constraints by [21, 12, 20] are based on this assumption. In their work, they learn a mapping from one domain to the other and back in a way that returns the original sample, which requires no loss of information. Nevertheless, this assumption fails to hold in a wide variety of domains. For example, in [21] they show that when learning a mapping from images of zebras to images of horses, the stripes of the zebras are lost, which results in an ambiguity when mapping in the other direction. In our paper, we do not make assumptions of this kind. Instead, we take a very generic formulation that fits a wide variety of domains.

A few publications suggested learning many to many transformations. These papers include the augmentation based extension of CycleGAN [1]. In their generative model, they provide an additional random vector for each domain. Other methods such as the NAM method [9] suggested non-adversarial training. In this model, the multiple solutions are obtained by different initializations. In our paper, multiple mappings are obtained by using a guide image.

A powerful method for capturing the relations between the two domains is done by employing two different autoencoders that share many of their parameters [16, 15]. These constraints provide a shared representation of the two domains. Low-level image properties, such as color, texture and edges are domain-specific and are encoded and decoded separately. The higher level properties are shared between the two domains and are processed by the same layers in the autoencoders. In our paper, we employ a shared encoder for both domains to enforce a shared representation. Each domain has its own separate encoder to encode domain-specific content. Weight sharing is not used.

bf Guided Translation

The most relevant line of work learns a mapping between the two domains that takes two images as inputs: a source image a from the first domain and a guide image b from the second domain [10, 14, 17, 18]. The work of [10, 14, 17] employ a very narrow encoding for the domain specific content that is reflected by a low dimensional encoding. This enables them to only encode the style of the image in their domain specific encoder. However, since this encoding is very limited, it is impossible to capture the entire domain specific content. In our method, we do not rely on architectural restrictions to partition the information in the images into domain specific and common parts. Instead, our losses provide sufficient and necessary conditions for dividing the content into domain-specific and common contents in a principled way. Therefore, in our method we are able to capture a disentangled representation in which the common information in its entirety is encoded in the shared encoder and the complete domain-specific information is encoded in the separate encoders.

The very recent work of [18] is probably the most similar to our work. In their paper, they tackle the problem where the source domain is a subset of the target domain (e.g., images of persons to images of persons with glasses). For such domains, a one-sided guided mapping from a source domain to a target domain is learned. For this purpose, they employ a common encoder, a separate encoder for the target domain and one decoder. To map between the source domain and the target domain, one applies the decoder on the common encoding of the source image and the separate encoding of the target domain. In their work, they are able to transfer the domain specific content for guiding the mapping from source to target. However, unlike our work, they are unable to handle the more general case, where both the source and the target domains have their own separate contents. This distinction is important, since even though they are able to provide content based guided mapping, they are limited to the case where the source domain behaves as a subset of the target domain. In our model, we are able to remove the content from the source images that is not present in the target images and not just to add content from images in the target domain.

Also related are several guided methods, which are trained in a supervised manner, i.e., the algorithm is provided with ground truth paired matches of images from domains A and B. Unlike the earlier supervised one-to-one mapping methods, such as pix2pix [11], these methods produce multiple outputs based on a guide image from the target domain. Examples include the Bicycle GAN by [22] and specific applications of the methods of [2, 6].

In our method, disentanglement between the shared content and the two sources of domain-specific information emerge. Other work that relies on unsupervised or weakly supervised disentanglement, include the InfoGAN method [4], which learns to disentangle a distribution to class-information and style, based on the structur of the data. [13, 7] learn a disentangled representation, by decreasing the class based information within it. We do not employ such class information.

2. Problem Setup

We consider a framework with two different visual domains $A = (\mathcal{X}_A, \mathbb{P}_A)$ and $B = (\mathcal{X}_B, \mathbb{P}_B)$. Here, $\mathcal{X}_A, \mathcal{X}_B \subset \mathbb{R}^n$ are two sample spaces of visual images and $\mathbb{P}_A, \mathbb{P}_B$ are two distributions over them (resp.), i.e., the probability of $x \sim \mathbb{P}_A$ being a is defined to be $\mathbb{P}_A[x = a]$.

In this setting, we have two independent training datasets $\mathcal{S}_A = \{a_i\}_{i=1}^{m_1}$ and $\mathcal{S}_B = \{b_j\}_{j=1}^{m_2}$ sampled i.i.d from $\mathbb{P}_A$ and $\mathbb{P}_B$ (resp.). The set $\mathcal{S}_A$ (resp. $\mathcal{S}_B$) consists of training images from domain A (resp. B).

Within a generative perspective, we assume that a sample $a \sim \mathbb{P}_A$ is distributed like $g(z_c, z_a, 0)$ and a sample $b \sim \mathbb{P}_B$ is distributed like $g(z_c, 0, z_b)$, where $z_c \sim \mathbb{P}_c$ and $z_a \sim \mathbb{P}_A^s$ and $z_b \sim \mathbb{P}_B^s$ are three latent variables. z_c is considered a *shared content* between the two domains and z_a and z_b

are *domain specific*. The process is subject to the following independency relations. A sample a from A is generated such that, $z_c \perp\!\!\!\perp z_a$ and a sample b from B is generated such that, $z_c \perp\!\!\!\perp z_b$. The function g takes a shared content $z_c \sim \mathbb{P}_c$ and a specific content $z_a \sim \mathbb{P}^s_A$ ($z_b \sim \mathbb{P}^s_B$) and returns an image $g(z_c, z_a, 0) \sim \mathbb{P}_A$ ($g(z_c, 0, z_b) \sim \mathbb{P}_B$). We assume that g is invertible for both domains, i.e., there are functions e^c, e^s_A and e^s_B, such that, for any sample $a \in \mathcal{X}_A$ and $b \in \mathcal{X}_B$, we have:

$$a = g(e^c(a), e^s_A(a), 0) \text{ and } b = g(e^c(b), 0, e^s_B(b)) \quad (1)$$

Here, e^c denotes the function that takes a sample a (or b) and returns its shared content, e^s_A takes a sample a and returns the specific content of a and e^s_B takes a sample b and returns its specific content. As mentioned above, $e^c(a) \sim e^c(b)$, $e^c(a) \perp\!\!\!\perp e^s_A(a)$ and $e^c(b) \perp\!\!\!\perp e^s_B(b)$. For clarity, we note this is just a matter of modeling and we do not assume knowledge of the distributions of z_c, z_a and z_b nor g, e^c, e^s_A and e^s_B.

As a running example, let A be a domain of images of non-smiling persons with glasses and B a domain of images of smiling persons without glasses. In this case, $\mathcal{X}_A$ is a set of images of persons with glasses, $\mathcal{X}_B$ is a set of images of smiling persons. In addition, $\mathbb{P}_A$ are $\mathbb{P}_B$ are two distributions over these sets (resp.). The set $\mathcal{S}_A$ consists of m_1 training images of persons with glasses and $\mathcal{S}_B$ consists of m_2 training images of smiling persons. Here, the shared content z_c between the two domains is an encoding of the identity and pose in an image (the image information excluding information about glasses or smile), z_a is an encoding of glasses and z_b is an encoding of a smile. The function g is a generator that takes an encoding z_c of a person and an encoding z_a of glasses (or an encoding z_b of a smile) and returns an image of the specified person with the specified glasses (or an image of the specified person with the specified smile).

In this paper, we aim to learn an encoder-decoder model $G \circ E(x)$. Our encoder E is composed of three parts: $E(x) := (E^c(x), E^s_A(x), E^s_B(x))$. Our goal is to make the first encoder, $E^c(x)$, capture the shared content between the two domains, $E^s_A(x)$, capture the content specific to images a from A and the third encoder, $E^s_B(x)$, capture the content present only in images b from B. In addition, we want to make our generator G be able to take $E^c(a)$ and $E^s_B(b)$ and return an image in B that has the shared content of a and the specific content of b (and similarly in the opposite direction). Both the encoder and decoder are implemented with neural networks of fixed architectures. The specific architectural details are given in the supplementary.

In the example above, for an image a from A, we would like $E^c(a)$ to encode the person in the image a (same for b from domain B). We also want $E^s_A(a)$ to encode the glasses in the image a and want $E^s_B(b)$ to encode the smile in the image b. We want G to take $E^c(a)$ and $E^s_B(b)$ and to return an image of the person in a without her glasses, but with the smile present in b.

Formally, we would like to have the following two properties on the encoder-decoder:

$$\begin{aligned} G(E^c(a), 0, E^s_B(b)) &\approx g(e^c(a), 0, e^s_B(b)) \\ \text{and } G(E^c(b), E^s_A(a), 0) &\approx g(e^c(b), e^s_A(a), 0) \end{aligned} \quad (2)$$

Here, 0 in the first equation stands for zeroing the coordinates of $E^s_A(x)$ in the encoder $E(x)$ (similarly for the second equation).

Since we do not have any paired matches of any of the forms: $(a, b) \mapsto g(e^c(a), 0, e^s_B(b))$ or $(a, b) \mapsto g(e^c(b), e^s_A(a), 0)$ (the left-hand-side is a pair of images and the right-hand-side is a single image) it is unclear how to make the encoder-decoder $G \circ E$ satisfy Eq. 2. Concretely, since we are only provided with unmatched images of persons with glasses and images of smiling persons, it is not obvious how to learn a mapping that takes an image of a person with glasses and an image of a smiling person and returns an image of the first person without the glasses, but with the smile from the second image. We present a set of training constraints that are both necessary and sufficient for performing this training.

3. Method

In Sec. 2 we defined the different components of the proposed framework. In addition, we explained that it is not obvious how to solve Eq. 2 without any supervised data. In this section, we explain our method for solving this problem in the proposed unsupervised setting.

As mentioned, our method consists of three encoders, E^c, E^s_A and E^s_B and a decoder G. E^c encodes the information content common to $\mathbb{P}_A$ and $\mathbb{P}_B$. The two other encoders, E^s_A and E^s_B, encode the information content specific to samples of $\mathbb{P}_A$ and $\mathbb{P}_B$ (resp.). To solve this, we use three types of losses: "zero", adversarial, and reconstruction.

3.1. Zero Loss

We would like to enforce E^s_A (E^s_B) to capture information relevant to domain A only. To do so we force E^s_A (E^s_B) to be 0 on samples in B (A):

$$\mathcal{L}^A_{zero} := \frac{1}{m_2} \sum_{j=1}^{m_2} \|E^s_A(b_j)\|_1 \quad (3)$$

$$\mathcal{L}^B_{zero} := \frac{1}{m_1} \sum_{i=1}^{m_1} \|E^s_B(a_i)\|_1 \quad (4)$$

$$\mathcal{L}_{zero} := \mathcal{L}^A_{zero} + \mathcal{L}^B_{zero} \quad (5)$$

As illustrated in Fig 1(a), if A is the domain of persons with glasses and B is that of smiling persons, then this loss ensures that E^s_A (E^s_B) will not capture any information about the face or smile (face or glasses).

3.2. Adversarial Loss

We would like to capture the fact that the common encoder, E^c, does not capture more information than necessary. In the running example, we would like E^c not to capture information about smile or glasses. This is illustrated in Fig 1(c). To do so, we use an adversarial loss to ensure that the distribution $\mathbb{P}_{E^c(A)}$ of $E^c(a)$ equals the distribution $\mathbb{P}_{E^c(B)}$ of $E^c(b)$. The loss $\mathcal{L}_{adv}$ is given by:

$$\frac{1}{m_1}\sum_{i=1}^{m_1} l(d(E^c(a_i)),1) + \frac{1}{m_2}\sum_{j=1}^{m_2} l(d(E^c(b_j)),1) \quad (6)$$

d is a discriminator network, and $l(p,q) = -(q\log(p) + (1-q)\log(1-p))$ is the binary cross entropy loss for $p \in [0,1]$ and $q \in \{0,1\}$. The network d minimizes the loss:

$$\mathcal{L}_d := \frac{1}{m_1}\sum_{i=1}^{m_1} l(d(E^c(a_i)),0) + \frac{1}{m_2}\sum_{j=1}^{m_2} l(d(E^c(b_j)),1) \quad (7)$$

The discriminator d attempts to separate between the distributions $\mathbb{P}_{E^c(A)}$ and $\mathbb{P}_{E^c(B)}$ of $E^c(a)$ and $E^c(b)$ (resp.), by classifying samples of the former as 0 and the samples of the latter as 1, whereas the encoder tries to fool the discriminator, hence forcing both distributions to match.

Referring back to our running example, this loss is a confusion term that ensures that the encoding by E^c of face images do not contain information on whether the person is smiling and on whether the person wears glasses.

3.3. Reconstruction Loss

Both the zero loss and the adversarial loss ensure that no encoder encodes more information than needed. However, we need to also ensure that all the needed information is encoded. In particular, E^s_A (E^s_B) should capture all the separate information in A (B). E^c should capture all the common information between A and B, but not less. To do so, we force the information in $E^s_A(a)$ and $E^c(a)$ to be sufficient to reconstruct a, and similarly that the information in $E^s_B(b)$ and $E^c(b)$ is sufficient to reconstruct b. Specifically, we have:

$$\mathcal{L}^A_{recon} := \frac{1}{m_1}\sum_{i=1}^{m_1} \|G(E^c(a_i), E^s_A(a_i), 0) - a_i\|_1 \quad (8)$$

$$\mathcal{L}^B_{recon} := \frac{1}{m_2}\sum_{j=1}^{m_2} \|G(E^c(b_i), 0, E^s_B(b_j)) - b_j\|_1 \quad (9)$$

$$\mathcal{L}_{recon} := \mathcal{L}^A_{recon} + \mathcal{L}^B_{recon} \quad (10)$$

3.4. Full Objective

For the full objective, E_c, E^s_A, E^s_B and G jointly minimize the following objective:

$$\mathcal{L} = \mathcal{L}_{zero} + \lambda_1\mathcal{L}_{adv} + \lambda_2\mathcal{L}_{recon} \quad (11)$$

Where λ_1 and λ_2 are positive constants. The discriminator d minimizes the loss $\mathcal{L}_d$ concurrently. The full description of the architecture employed for the encoders, generator and discriminator is given in the supplementary material.

4. Theoretical Analysis

We provide an informal theoretical analysis for the success of the proposed method. For the formal version, please refer to the supplementary material.

In Sec. 2 we represented our random variable $a \sim \mathbb{P}_A$ and $b \sim \mathbb{P}_B$ in the following forms $a = g(e^c(a), e^s_A(a), 0)$ and $b = g(e^c(b), 0, e^s_B(b))$, where $e^c(a) \perp\!\!\!\perp e^s_A(a)$, $e^c(b) \perp\!\!\!\perp e^s_B(b)$ and g is an invertible function.

Before we present our theorem regarding emerging disentanglement between the learned encoders, we provide a necessary definition of an intersection. An intersection of two independent random variables a and b are two representations $a = g(e^c(a), e^s_A(a), 0)$ and $b = g(e^c(b), 0, e^s_B(b))$, such that, the common encoding $e^c(a) \sim e^c(b)$ has the largest amount of information (measured by entropy H). For example, let us consider the case in which domain A consists of images of persons wearing glasses and domain B consists images of smiling persons. In this case, we can encode the samples of A into (i) an identity and pose encoding and (ii) a glasses encoding. Similarly, we can encode the samples of B into the first encoding of domain A and the encoding of the smile. This representation forms an intersection, since we cannot transfer common information from the glasses and the smile into the common part.

Definition 1 (Intersection)**.** *We say that the two representations $a = g(e^c(a), e^s_A(a), 0)$ and $b = g(e^c(b), 0, e^s_B(b))$ form an intersection between a and b, if for any other representation $a = \hat{g}(\hat{e}^c(a), \hat{e}^s_A(a), 0)$ and $b = \hat{g}(\hat{e}^c(b), 0, \hat{e}^s_B(b))$, such that, $\hat{g}$ is invertible and $\hat{e}^c(a) \sim \hat{e}^c(b)$, we have: $H(\hat{e}^c(a)) \leq H(e^c(a))$.*

The following theorem shows that under reasonable conditions, by minimizing the proposed losses, we obtain a disentangled representation.

Theorem 1 (Informal)**.** *In the setting of Sec. 2. Let $a \sim \mathbb{P}_A$ and $b \sim \mathbb{P}_B$ be two random variables. Assume that the representations $g(e^c(a), e^s_A(a), 0)$ and $g(e^c(b), 0, e^s_B(b))$ form an intersection between a and b. Assume that we cannot recover the sample a from the separate encoding $E^s_A(a)$. Assume that the reconstruction and adversarial losses are minimized by E^c, E^s_A, E^s_B and G. Then, we obtain a disentanglement between $E^c(a)$ and $E^s_A(a)$, such that, $E^c(a)$ captures the information of $e^c(a)$ and $E^s_A(a)$ captures the information of $e^s_A(a)$.*

The theorem makes three types of assumptions. The first type is about the modeling of the data, i.e., that it follows the

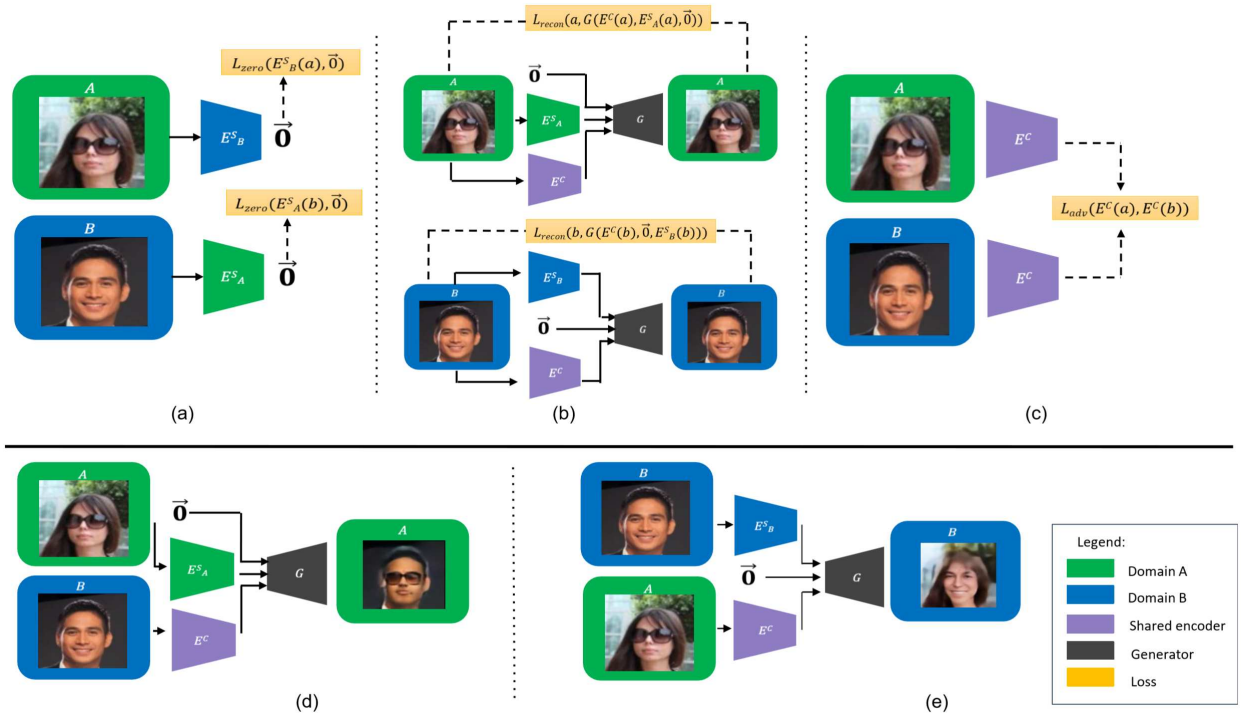

Figure 1. Illustration of the train and inference stages. The losses are illustrated in (a), (b) and (c) and the guided mappings are illustrated in (d) and (e). (a) Illustration of the zero loss. Encoding images from domain A (illustrated in green) with domain's B separate encoder should result in a zero vector, encoding no information about the image (and vice versa). (b) Illustration of the reconstruction loss. Given a's separate encoding (illustrated in green), for example glasses, and its common encoding (illustrated in purple), for example all other facial features, it should be possible to reconstruct a (same for domain B). (c) Illustration of the adversarial loss. The distribution of the common encoding from domain A and domain B (face features) should be the same. To enforce this, an adversarial loss is used. (d) Constructing new images. At inference time we can encode domain's B image b using its separate encoder to get its smile, encode the common domain A's image a (face features without glasses) and generate an image similar to a, but without glasses and with b's smile. (e) Similarly to (d), we can generate an image similar to a but with the smile removed and glasses of b added.

problem definition in Sec. 2 and that the shared part of the model (e^c) is an intersection of the two domains. The second assumption is regarding the separate encoder we learn (E^s_A) and it states that one cannot reconstruct a from $E^s_A(a)$. The last group of assumptions concerns the losses, which we minimize in our algorithm.

The conclusion of this theorem is that under the proposed assumptions, (i) the common $E^c(a)$ and separate $E^s_A(a)$ parts are independent, (ii) the common part $E^c(a)$ captures the information in the underlying $e^c(a)$, and (iii) the separate part $E^s_A(a)$ captures the information in $e^s_A(a)$. Therefore, we obtain the desired encoding of domain A. By symmetric arguments, we arrive at the same conclusions for $E^c(b)$ and $E^s_B(b)$.

5. Experiments

To evaluate our method, we consider the celebA [19] dataset, which consists of celebrity face images with different attributes. We consider the smile, glasses, facial hair, male, female, blond and black hair attributes. Each of these attributes can be used as domain A or B symmetrically.

5.1. Guided translation between domains

In Fig. 2, we consider A to be the domain of images of smiling persons and B to be the domain of images of persons with glasses. Given a sample $a \in A$ (top row) and a sample $b \in B$ (left column), each image constructed is of the form $G(E^c(a), 0, E^s_B(b))$. The common features of image a (its identity) are preserved, the smile is removed, and the glasses of b are added (the guide image). The reverse direction, as well as other cross domain translations, are depicted in the supplementary.

In order to evaluate the success of the translation numerically, we pretrain a classifier to distinguish between images from domain A and domain B. If the specific part of the domain A was successfully removed (for example, smile), and the specific part of domain B was successfully added (for example, glasses), then the classifier should classify the translated image as a domain B image. Tab. 1 shows the success of our method in this case, in comparison to

	Smile To Glasses	Glasses To Smile	Facial Hair To Smile	Smile To Facial Hair	Facial Hair To Glasses	Glasses To Facial Hair
Fader networks [13]	76.8%	97.3%	95.4%	84.2%	77.8 %	85.2%
Guided content transfer [18]	45.8%	92.7%	85.6%	85.1%	38.6%	82.2%
MUNIT [10]	7.3%	9.2%	9.3%	8.4%	7.3%	8.5%
DRIT [14]	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.

	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 [18]	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 [18]	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 [18]	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.

the baseline methods of [18, 13, 10, 14], which are much less successful in switching attributes. Specifically: (i) MUNIT [10] and DRIT [14] only change style, but the content is unchanged, (ii) Fader networks [13] translated between the domains, in a less convincing way, that also ignores the guide image, and (iii) The method of Press et al. [18] adds the element of the target domain, but fails to remove the content of the source domain.

By conducting a user study, we evaluate the ability to (a) remove the specific attribute of domain A (b) add the specific attribute of domain B, and (c) preserve the identity of the image encoded in the common encoder. To do so, given an image a from domain A and an image b from domain B, we present the user with two images $a \in A$, $b \in B$ and the generated image $G(E^c(a), 0, E^s_B(b))$ (or $G(E^c(b), E^s_A(a), 0)$ for the reverse direction), and ask the following three questions: 1. Is the specific attribute of A (e.g smile) removed? 2. Is the guided image b specific attribute (e.g glasses) added? 3. Is the identify of a's image preserved (that is, is the common attribute from a still present in the image)? Mean Opinion Score on the scale of 1 to 5, are collected for 20 randomly selected test images in A and B by 20 different users is reported in Tab. 2. For most translations, the ability to remove A's specific attribute and add B's specific attribute is significantly better than that of [18], while the ability to preserve the identity of a is on-par with [18]. The Fader networks [13] provides a generic (unguided) cross domain translation, and MUNIT [10] transfers style and not content and were therefore not included in the user study. See supplementary for the results obtained by these methods.

5.2. Linearity of latent space

We evaluate the linearity of the latent representation of A's separate encoder, B's separate encoder and the common encoder. In this case, A serves as the domain of images of smiling persons and B of images of persons with facial hair. In Fig. 3 the generated images take the form $G(com, a, 0)$, where $com = \alpha E^c(a_1) + (1-\alpha)E^c(a_2)$ and $a = \beta E^s_A(a_3) + (1-\beta)E^s_A(a_4)$. α ranges between 0 and 1, going left to right and β ranges from 0 to 1, going from top to bottom. a_1, a_2, a_3, a_4 are images from domain A (smiling persons), given in the top row and left column. We observe that the latent representations produced by A's separate encoder and the common encoder are linear.

Similarly, in Fig. 4 we evaluate the linear separability of B's separate encoder. Generated images take the form $G(com, 0, b)$, where $com = \alpha E^c(a_1) + (1-\alpha)E^c(a_2)$ and $b = \beta E^s_B(b_1) + (1-\beta)E^s_B(b_2)$. α ranges between 0 and 1, going left to right, and β ranges between 0 and 1, going from top to bottom. a_1, a_2 are images from domain A given in the top row and b_1, b_2 are images from domain B in the left column.

Lastly, in Fig. 5, we fix the common part from some image c, and evaluate the linearity of both separate encoders applied together. Generated images take the form $G(com, a, b)$, where $com = E^c(c)$ and $a = \alpha E^s_A(a_1) + (1-\alpha)E^s_A(a_2)$ and $b = \beta E^s_B(b_1) + (1-\beta)E^s_B(b_2)$. α

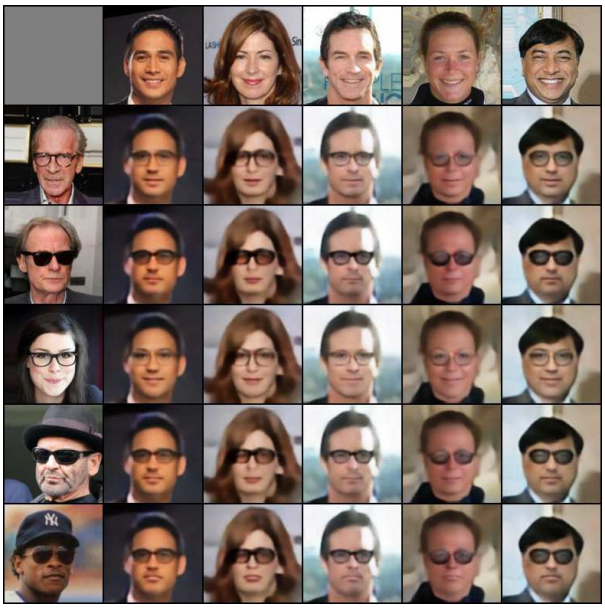

Figure 2. Images $a \in A$ are in the top row and $b \in B$ in the left column. The images constructed are $G(E^c(a), 0, E^s_B(b)))$, consisting of the common parts of a and separate part of b (smile is removed and glasses added).

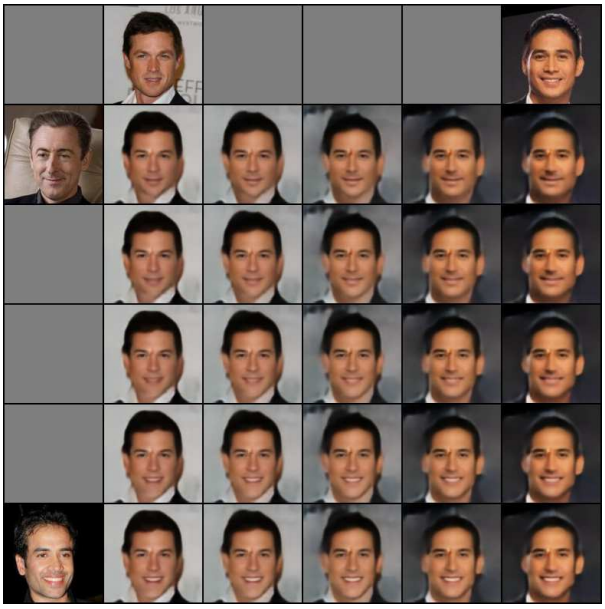

Figure 3. Interpolation in the latent space of domain A (smiling). We linearly interpolate between the common encoding of the two images in the top row going left to right. Concurrently, we linearly interpolate between the separate encoding of the two images in the left column going top to bottom.

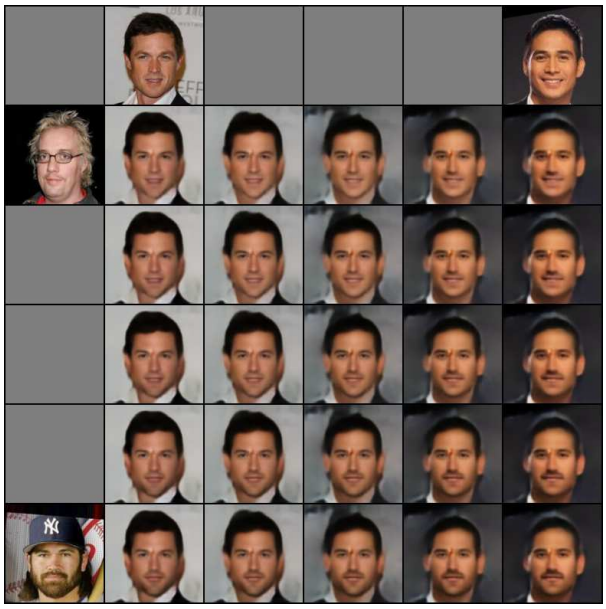

Figure 4. Interpolation in the latent space of domains A (smiling) and B (facial hair). We interpolate the common encoding of the two images from domain A in the top row. Concurrently, we linearly interpolate between the separate encoding of the two images from domain B in the left column.

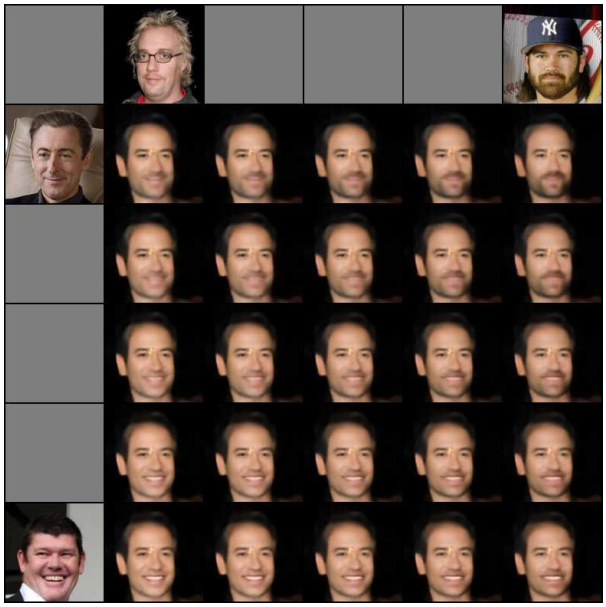

Figure 5. Interpolation domains A (smiling) and B (facial hair). Fixing the common encoding to randomly chosen image, we interpolate between A's separate encoding of the two images in the top row. Concurrently, we interpolate between B's separate encoding of the two images in the left column.

ranges from 0 to 1 going left to right and β ranges from 0 to 1 going from top to bottom. c is a fixed image in A, while a_1, a_2 are images from domain A given in the top row and b_1, b_2 are images from domain B in the left column.

Note that in this last case, we generate images from the union domain, i.e., create images that have, in addition to the common information, both the added content of A and of B. The method also allows us to consider the intersec-

Figure 6. Generating images from the intersection of A and B. (top) image from A. (bottom) mapping to the intersection domain.

tion domain. In the depicted example, domain A includes images of persons with glasses and B includes images of smiling persons. The intersection of A and B consists of images of non-smiling persons (without glasses). Having never seen such images in the training set, our method now allows us to generate images from this distribution. This is illustrated in Fig. 6. To do so, the generated image is of the form $G(E^c(x), 0, 0)$, where x is a member of A or B.

5.3. Unsupervised Domain Adaptation

To evaluate the disentangled representation, we perform unsupervised domain adaptation experiments translating from MNIST to SVHN. In this problem, the underlying framework is used to translate from MNIST to SVHN and a pretrained classifier is used to evaluate the percentage of images mapped to the same label in the target domain. In our case, given an MNIST digit a, we randomly sample an SVHN digit b and consider the translation to SVHN as $G(E^c(a), 0, E^s_B(b))$. In the MNIST to SVHN direction our method has 61.0% accuracy beating Vae-NAM [8] (51.7%), NAM [9] (31.9%), DistanceGAN [3] (27.8%) and CycleGAN [21] (17.7%). In the reverse direction it has 41.0% accuracy beating Vae-NAM (37.4%), NAM (33.3%), DistanceGAN (27.8%) and CycleGAN (26.1%).

5.4. Ablation study

We consider the formulation of our objective with each of the three parts missing: the adversarial loss, the zero loss and the reconstruction loss. We conduct an ablation study in the case of A being images of smiling persons and B is the domain of images of persons with glasses. The results, which appear in Tab. 3 and shown visually in the supplementary, indicate that when the reconstruction loss is missing, the method is unable to generate realistic looking images. In the case of no adversarial loss, the method is able to remove the smile but unable to add glasses from b. Without the adversarial loss, the common encoder can contain information specific to the domain, such as glasses, and so there would be no need to encode it in the separate encoder. Lastly, without the zero loss, the translation is slightly worse but still succeeds to a large extent. As shown in our analysis, the enforcing of the zero loss is not required to achieve the desired disentanglement effect.

All Losses	91.8%	99.3%
No zero loss	85.4%	97.8%
No adversarial loss	64,5%	79.3%
No reconstruction loss	50.0%	50.0%

Table 3. An ablation study for the case where A is persons with glasses and B is smiling persons. We consider the same setting as Tab 1, and consider the effect of removing each loss on the classification loss. The left column is for the Smile To Glasses task and the right column is for the Glasses To Smile task.

6. Conclusions

The field of unsupervised learning presents new problems that go beyond the classical methods of clustering or density estimation. The problem of unsupervised cross-domain translation was not considered solvable up to a few years ago. Recently, a set of guided translation problems have emerged, in which one maps between domains based on the features of a reference image in the target domain. While the literature methods treat the two domains in an asymmetric way (one domain donates style and another content, or one domain is a subset of the second), our work is the first to treat the domains in a symmetric way.

Our work also presents the first method that is able to create images that have guided elements from two different domains, extracted from donor images a and b (one from each domain) and overlaid on a third image (taken from either domains) that donates the shared content.

The method we propose is shown to provide a sufficient set of constraints in order to support this conversion. It does not employ GANs in the visual domains, or cycles of any sort. The constraints are simple structural and reconstruction constraints, with the addition of a domain confusion loss, applied in the shared latent space.

Our experiments show that the new method provides superior results for the symmetrical guided domain problem in comparison to the literature methods. Going forward, the ability to intersect domains (creating a domain that is orthogonal to the specific parts of the two domains), construct their union (combining both specific parts and the shared part), and consider the difference between the two, could lead to the ability to perform domain arithmetics and construct complex visual domains by combining, in a very flexible way, an unlimited number of domains.

Acknowledgements

This project has received funding from the European Research Council (ERC) under the European Unions Horizon 2020 research and innovation programme (grant ERC CoG 725974). The contribution of Sagie Benaim is part of a Ph.D. thesis research conducted at Tel Aviv University.

References

[1] Amjad Almahairi, Sai Rajeshwar, Alessandro Sordoni, Philip Bachman, and Aaron Courville. Augmented CycleGAN: Learning many-to-many mappings from unpaired data. In *ICML*, 2018. 2

[2] Jianmin Bao, Dong Chen, Fang Wen, Houqiang Li, and Gang Hua. Cvae-gan: Fine-grained image generation through asymmetric training. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2764–2773. IEEE, 2017. 2

[3] Sagie Benaim and Lior Wolf. One-sided unsupervised domain mapping. In *NIPS*, 2017. 8

[4] Xi Chen, Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets. In *NIPS*. 2016. 2

[5] Tomer Galanti, Lior Wolf, and Sagie Benaim. The role of minimal complexity functions in unsupervised learning of semantic mappings. In *International Conference on Learning Representations*, 2018. 1

[6] Abel Gonzalez-Garcia, Joost van de Weijer, and Yoshua Bengio. Image-to-image translation for cross-domain disentanglement. In *NIPS*, 2018. 2

[7] Naama Hadad, Lior Wolf, and Moni Shahar. A two-step disentanglement method. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 772–780, 2018. 2

[8] Yedid Hoshen. Non-adversarial mapping with vaes. In *Proceedings of the 32Nd International Conference on Neural Information Processing Systems*, NIPS'18, pages 7539–7548, USA, 2018. Curran Associates Inc. 8

[9] Yedid Hoshen and Lior Wolf. NAM - unsupervised cross-domain image mapping without cycles or GANs. In *ICLR workshop*, 2018. 2, 8

[10] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *ECCV*, 2018. 1, 2, 6

[11] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017. 2

[12] Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jungkwon Lee, and Jiwon Kim. Learning to discover cross-domain relations with generative adversarial networks. *arXiv preprint arXiv:1703.05192*, 2017. 2

[13] Guillaume Lample, Neil Zeghidour, Nicolas Usunier, Antoine Bordes, Ludovic Denoyer, et al. Fader networks: Manipulating images by sliding attributes. In *NIPS*, pages 5967–5976, 2017. 2, 6

[14] Hsin-Ying Lee, Hung-Yu Tseng, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Diverse image-to-image translation via disentangled representations. In *The European Conference on Computer Vision (ECCV)*, September 2018. 1, 2, 6

[15] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. In *NIPS*. 2017. 2

[16] Ming-Yu Liu and Oncel Tuzel. Coupled generative adversarial networks. In *NIPS*, pages 469–477. 2016. 2

[17] Liqian Ma, Xu Jia, Stamatios Georgoulis, Tinne Tuytelaars, and Luc Van Gool. Exemplar guided unsupervised image-to-image translation. *arXiv preprint arXiv:1805.11145*, 2018. 2

[18] Ori Press, Tomer Galanti, Sagie Benaim, and Lior Wolf. Emerging disentanglement in auto-encoder based unsupervised image content transfer. In *International Conference on Learning Representations*, 2019. 1, 2, 6

[19] Shuo Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. From facial parts responses to face detection: A deep learning approach. In *ICCV*, pages 3676–3684, 2015. 5

[20] Zili Yi, Hao Zhang, Ping Tan, and Minglun Gong. DualGAN: Unsupervised dual learning for image-to-image translation. *arXiv preprint arXiv:1704.02510*, 2017. 2

[21] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networkss. *arXiv preprint arXiv:1703.10593*, 2017. 2, 8

[22] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. Toward multimodal image-to-image translation. In *NIPS*, 2017. 2