ISF-GAN: An Implicit Style Function for High-Resolution Image-to-Image Translation

Yahui Liu, Yajing Chen, Linchao Bao, Nicu Sebe, Senior Member, IEEE, Bruno Lepri, and Marco De Nadai

Abstract—Recently, there has been an increasing interest in image editing methods that employ pre-trained unconditional image generators (e.g., StyleGAN). However, applying these methods to translate images to multiple visual domains remains challenging. Existing works do not often preserve the domain-invariant part of the image (e.g., the identity in human face translations), or they do not usually handle multiple domains or allow for multi-modal translations. This work proposes an implicit style function (ISF) to straightforwardly achieve multi-modal and multi-domain image-to-image translation from pre-trained unconditional generators. The ISF manipulates the semantics of a latent code to ensure that the image generated from the manipulated code lies in the desired visual domain. Our human faces and animal image manipulations show significantly improved results over the baselines. Our model enables cost-effective multi-modal unsupervised image-to-image translations at high resolution using pre-trained unconditional GANs. The code and data are available at: https://github.com/yhlleo/stylegan-mmuit.

Index Terms—Face editing, generative adversarial networks (GANs), unsupervised image-to-image translation.

I. INTRODUCTION

Generative methods have become increasingly effective at synthesizing realistic images at high resolution, stimulating new practical applications in academia and industry. Many different tasks have been proposed, from super-resolution [1], [2] to image manipulation [3], [4], from image-to-image and text-to-image translations [5]–[8] to video generation [9], [10]. Yet, training these task-specific models at high resolution (e.g., 1024 × 1024) is very computationally expensive. For this reason, recent works have interpreted and exploited the latent space of pre-trained high-resolution unconditional Generative Adversarial Networks (GANs) to solve several generative tasks without training a generator from scratch [11]–[17]. Most of these approaches are based on StyleGAN ([18], [19]), the state-of-the-art of unconditional image generation. StyleGAN maps a noise vector $z \sim \mathcal{N}(0, I_n)$ to an intermediate and learned latent space $\mathbf{W}^+$, which exhibits some intriguing disentangled semantic properties [20]–[22] that can be interpreted and exploited. For example, InterFaceGAN [23], [24] identifies the semantic attributes in $\mathbf{W}^+$ to manipulate the semantics of a latent code and change the facial attributes of images. However, existing models manipulating StyleGAN latent codes allow to edit just one attribute per time or fail to preserve the content of the image not involved in the manipulation, resulting, for example, in eye color changes when gender is manipulated (see Fig. 1). Moreover, most existing models are deterministic (i.e., not multi-modal). Altogether, these issues make existing approaches unsuitable for Multi-modal, and Multi-domain Unsupervised Image-to-image Translation (MMUIT) [6], [25].

An ideal MMUIT model should be able to change the domain-specific parts of the image while preserving the content of the image (e.g., the background and the identity of people) and...
synthesize images considering the multiple appearance modes within each domain [6], [25]. Here, domain refers to a set of images having some distinctive visual pattern, usually called style. For example, we can group images based on the gender of people in pictures (see Fig. 1). Since paired images (e.g., pictures of the same person with different gender) are usually not available, the model should be unsupervisedly trained [5], [6], [25], [26].

This paper proposes an implicit style function (ISF) that allows leveraging a pre-trained unconditional image generator to do MMUITs. The proposed method works directly on the latent space and enables the change of multiple semantic attributes at a time without affecting any content of the image that is not involved in the translation. Given a latent code in the latent space of pretrained GANs (e.g., \( \mathbf{W}^+ \) in StyleGAN [18]), also called style code, \( \mathbf{w} \), a semantic attribute label \( \mathbf{d} \), and a noise vector \( \mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \), the ISF outputs a new style code \( \mathbf{w}^* \) that contains the desired semantic attributes. By randomly sampling multiple \( \mathbf{z} \), several variants of the desired image translation (e.g., multiple hairstyles in gender manipulations) can be generated.

As a consequence, our proposed method can be used for multiple tasks, including MMUIT [6], [25], [27] and smooth semantic interpolation [25], [28]. When used in conjunction with GAN inversion techniques, which find the most similar style code given an input image (e.g., [12], [29]), our proposed method can also translate real images at high-resolution.

To enable better manipulations of StyleGAN’s latent codes, we propose a simple extension of Adaptive Instance Normalization (AdaIN) [30], namely Adaptive Layer Normalization (AdaLN). AdaLN normalizes the latent code through Layer Normalization (AdaIN) [30], namely Adaptive Layer Normalization (AdaLN). AdaLN normalizes the latent code through Layer Normalization [31] instead of Instance Normalization [32] and goes beyond the assumption of independence between channels by computing the normalization for each sample across all the channels. Quantitative and qualitative results in face manipulations show that our proposed method outperforms state-of-the-art methods.

Our contributions can be summarized as follows:

- We propose an implicit style function for editing multiple semantics of the input image at once by StyleGAN latent space manipulations. The images generated from the edited latent codes preserve the content/identity of input images while changing the user-specified semantics;
- We propose AdaLN, a simple extension of the popular normalization method AdaIN [30] to enable better arbitrary style transfer in the StyleGAN latent space;
- At the best of our knowledge, we are the first to address multi-domain and multi-modal unsupervised image-to-image translation by a pre-trained and fixed image generator, which allows having image translations at high resolution (i.e., \( 1024 \times 1024 \)).

II. RELATED WORK

Unsupervised MMUIT models aim at learning a mapping between different visual domains unsupervisedly. They require that: 1) the domain-invariant part of the image, also called content, is preserved; 2) the model can map multiple domains; and 3) the output is multi-modal (i.e., diverse translations can be generated with the same input image).

To the best of our knowledge, there are no works, based on pre-trained and fixed unconditional GANs, that simultaneously meet all the requirements of unsupervised MMUIT models. However, we here review the literature of MMUIT models, StyleGAN-based latent space manipulations and StyleGAN-based inversion techniques.

Image-to-image Translation: Early attempts in Image-to-Image Translation are based on paired images [33]–[36] and one-to-one domain mappings [26], [37]–[41]. More recently, many studies have focused on training a single model that maps images into multiple domains and overcomes the deterministic (i.e., one-to-one) translation often assumed by previous works (e.g., [5], [39], [42]). We refer to these methods as MMUIT models. For example, DRIT++ [27] assumes the existence of a domain-independent (“content”) and a domain-specific (“style”) image representation and obtain this using two separate encoders. Then, DRIT++ allows multi-modal translations injecting random noise in the generation process. GMM-UNIT [8], [43] uses a Variational Auto-Encoder approach [44] where a style encoder maps the image into a Gaussian Mixture Model, from which it is possible to sample multiple different styles to be used in the generation process. StarGAN v2 [6], [25] proposes an architecture composed by a style encoder, which maps an existing image to a style code, and a mapping function, which starts from random noise and a domain code to generate a style code. In doing so, they achieve state-of-the-art performance in high-resolution (i.e., \( 256 \times 256 \)) and diversity of translated images.

The current state-of-the-art models in MMUIT field show results usually at \( 256 \times 256 \) resolution and require training a generator from scratch. We here aim at enabling high-resolution MMUIT without training a generator.

StyleGAN Latent Space Manipulation: Most conditional GANs are trained at low and medium resolutions (i.e., up to \( 256 \times 256 \) pixels). Scaling the image generators to higher resolutions to solve specific tasks requires a substantial computational budget. For this reason, scholars have recently started exploring and interpreting the latent code semantics of large unconditional networks for image synthesis such as PGGAN [45] and StyleGAN [18], [19]. The general idea is to adapt unconditional GANs to solve specific tasks at high resolution without training task-specific generators [16]. For example, Härkönen et al. [21] identify important latent directions based on Principal Component Analysis (PCA) to control properties such as lighting, facial attributes, and landscape attributes. Shen et al. [24] use an off-the-shelf classifier to find the linear hyperplanes of semantic facial attributes. InterFaceGAN [23], [24] learns the latent semantics through linear hyper-planes, which rely on multiple SVMs that have to be specifically trained on each domain translation. Then, it edits face images moving the latent code with linear transformations. PSP [29] learns an encoder and a mapping function that enables multi-modal domain translations but does not support multi-domain translation. Chai et al. [46] propose to learn an imperfect masked encoder that finds the latent code that best resembles the input masked or coarsely-edited image. Then, it uses the fixed generator to synthesize photo-realistic results showing applications in image composition, image inpainting.
and multi-modal editing. Finally, StyleFlow [13] proposes to have attribute-controlled sampling and attributed-controlled editing through StyleGAN. They learn a function \( \Phi(z, d) \) to sample StyleGAN latent codes conditioned to a semantic attribute vector, and they allow editing specific blocks of the StyleGAN latent space to edit pre-defined semantic categories.

However, existing latent space manipulation models based on StyleGAN are not suitable for MMUIT tasks. They are either limited on single-attribute at a time (e.g., [23], [24]) or one-domain translations (e.g., [29]). Moreover, they have very limited if no diversity in the manipulations (e.g., [23], [24], [29]) and, more importantly, manipulations along one attribute can easily result in unwanted changes along with other attributes (e.g., face features changes in hair color manipulations). We propose a simple and effective Implicit Style Function to manipulate the StyleGAN latent code and enable MMUIT at high resolution with a pre-trained and fixed StyleGAN model.

**GAN inversion in StyleGAN latent space:** Solving the GAN inversion problem is essential to use pre-trained GANs with real images. GAN inversion techniques aim to find the generator’s latent code that best corresponds to a given image.

Existing inversion approaches typically fall into two categories: encoder-based and optimization-based methods. The former [29] uses LBFGS [47] or similar optimizers to find the latent code \( z \) that best recovers the image \( x \) with \( z^* = \text{arg min}_z (\text{dist}(G(z), x)) \) where \( \text{dist} \) is a metric distance function in the image space. The latter models (e.g., [11], [12], [48]) instead speeds-up the process at inference time by learning an encoder \( E \). Then, the optimal \( z \) is simply the result of a feed-forward pass through \( E \), more formally: \( z^* = E(x) \).

**Fusion methods:** A noteworthy application of Image-to-image Translation is image fusion, which aims to exploit images obtained by different sensors to generate better and more robust images. For example, FusionGAN [49] focuses on fusing the thermal radiation information in infrared images and the texture detail information in visible images to generate images through a GAN. Pan-GAN [50] instead focuses on fusing the thermal radiation information in infrared images and the texture detail information in visible images to generate images through a GAN. The latter focuses on remote sensing images fusing different layers of information to do pan-sharpening and better preserve the spectral and spatial information in images.

### III. Method

An unconditional generator \( G : \mathbf{Z} \rightarrow \mathbf{X} \) learns to synthesize an image \( x \in \mathbf{X} \) given a vector \( z \in \mathbf{Z} \), lying in a low-dimensional latent space. In this work, we focus on StyleGAN v1 [18] and StyleGAN v2 [19], which first learn a non-linear mapping \( f : \mathbf{Z} \rightarrow \mathbf{W}^+ \) that outputs \( w = f(z) \in \mathbb{R}^{18 \times 512} \), and then generates the image from \( \mathbf{W}^+ \) space: \( x = G(w) \).

We aim at manipulating \( w \) such that it follows some user-specified semantic attributes \( d \in \mathbf{D} \subset \mathbb{R}^m \) that refer to a \( m \)-dimensional semantic space, while best preserving the attribute-invariant content (e.g., face identity) of the source images. In other words, we want to learn a non-linear neural Implicit Style Function (ISF) \( \mathcal{M} : \mathbf{W}^+ \rightarrow \mathbf{W}^+ \) that outputs a manipulated code \( w^* \) in which:

1. **The output looks realistic:** The generated image \( x^* = G(w^*) \) should look as realistic as the images generated from non-manipulated latent codes;
2. **Generated images exhibit the desired semantics:** The translation from \( w \) to \( w^* \) should be semantically correct;
3. **The source content is preserved:** A translated image \( x^* = G(w^*) \) should maintain the attribute-invariant characteristics of the image and change only the attribute-specific properties of the source image \( x \). For example, in smile \( \leftrightarrow \) non-smile translations, \( x^* \) should maintain the pose and identity of the source image \( x \);
4. **The output is multi-modal:** Translations are inherently ambiguous, thus it is desirable to have a model that generates different, plausible translations. Note that we are not here interested in images whose appearance is almost identical but e.g. individual hairs are placed very differently (such in StyleGAN [18]);
5. **Training is multi-domain:** A single function \( \mathcal{M} \) should be able to map each style code into multiple, different domains.

Simultaneously meeting these properties allows us to overcome several limits of existing approaches and enable MMUITs using pre-trained and frozen unconditional GANs.

To fulfill the above properties, we propose a multilayer perceptron (MLP) as Implicit Style Function \( \mathcal{M} \), which can be formulated as:

\[
    w^* = \mathcal{M}(w, z, d),
\]

where \( \mathcal{M}(\cdot) \) predicts a target latent code \( w^* \) from a source latent code \( w \), a noise vector \( z \) randomly sampled from a standard Gaussian (i.e., \( z \sim \mathcal{N}(0, I_n) \)), and a target semantic vector \( d \). The source latent code \( w \) represents the latent code to be manipulated and \( z \) ensures the stochasticity of the manipulation process.

In order to learn such \( \mathcal{M} \) in an unsupervised scenario, we assume to have a discriminator at training time which consists of two sub-tasks: (1) a binary classifier to distinguish the generated images from realistic images, and (2) a multi-label classifier \( f : \mathbf{X} \rightarrow \mathbf{D} \) to distinguish whether the images are with expected visual semantics. Thus, the \( \mathcal{M} \) is learnt in an adversarial approach.

### A. Learning the Implicit Style Function

We propose to train our ISF \( \mathcal{M} \) in an adversarial fashion [51] with the help of a multi-task discriminator \( D \), which is learned together with \( \mathcal{M} \). At training time, we only optimize the parameters in the networks \( \mathcal{M} \) and \( D \), while we do not update the parameters of the pre-trained generator \( G \). Such a strategy helps our method requiring fewer resources than traditional models that train a generator from scratch. The detailed architecture is depicted in Fig. 2.

We use several objective functions to constraint the training process and learn the ISF \( \mathcal{M} \).
Realism & Semantic Correctness: Inspired by recent literature (e.g., [5], [25]), we use a discriminator $D$ with two branches namely $D_{\text{cls}}$, devoted to domain classification, and $D_{\text{ds}}$, which learns a binary classification determining whether an image $x$ is real or fake. More formally, we ensure the realism of the synthesized images through:

$$
\mathcal{L}_{\text{iff}} = \mathbb{E}_{w,z,d} \left[ \log D_{\text{iff}}(G(w)) + \log(1 - D_{\text{iff}}(G(M(w, z, d)))) \right].
$$

To ensure a manipulated code $w^* = M(w, z, d)$ generates an image with the specified semantics $d$, we impose a domain classification loss when optimizing $D$ and $M$. More formally, $D$ calculates the cross-entropy loss on the generated images over the non-manipulated latent codes by:

$$
\mathcal{L}_{\text{D}}^{\text{M}} = \mathbb{E}_{w,d_0} \left[ - \log D_{\text{cls}}(d_0 | G(w)) \right],
$$

where $d_0 = f(G(w))$ refers to the attribute vector extracted from the image generated from $G(w)$ by utilizing the pre-trained attribute classifiers $f$. We will explain $f$ in the Experiments Section. To learn classifying images, $M$ tries to minimize the classification error of images generated with manipulated latent codes with:

$$
\mathcal{L}_{\text{cls}}^{\text{M}} = \mathbb{E}_{s,z,d} \left[ - \log D_{\text{cls}}(d | G(M(s, z, d))) \right].
$$

Content Preservation: We propose to preserve the content during the latent code manipulation with the following loss:

$$
\mathcal{L}_{\text{cont}} = \mathbb{E}_{s,z,d} \left[ \psi(G(s), G(M(s, z, d))) \right],
$$

where $\psi(x_1, x_2)$ estimates the perceptual distance between image $x_1$ and image $x_2$ using an externally pre-trained network. Eq. (5) minimizes the perceptual distance between $G(w)$ and any image generated with a manipulated code (i.e., $G(M(w, z, d)))$. Similarly to Liu et al. [25], Eq. (5) implies that some perceptual features should be maintained during the manipulation process. Although different perceptual distances can be used (e.g., the $L_2$ distance on VGG features [52]), we implement $\psi(x_1, x_2)$ using the Learned Perceptual Image Patch Similarity (LPIPS) metric [53], which has been proved to be better aligned with the human perceptual similarity.

To further help the network to preserve the content of the input image, we also introduce a neighboring constraint:

$$
\mathcal{L}_{\text{nb}} = \mathbb{E}_{w,z,d} \left[ \| w - M(w, z, d) \|_2 \right].
$$

Eq. (6) is motivated by previous literature that has shown neighboring latent codes in StyleGAN exhibit similar semantic properties [18], [19].

Finally, we also encourage the cycle consistency [5], [6], [37], which is typically used in most MMUIT models, to stabilize the training process. The cycle consistency loss encourages the image generated by the original latent code $w$ and the image synthesized by the latent code mapped back into the original domain $M(w^*, z, d_0)$ to be as similar as possible. More formally, the cycle consistency loss is:

$$
\mathcal{L}_{\text{cyc}} = \mathbb{E}_{w,z,d,d_0} \left[ \| G(w) - G(M(w^*, z, d_0)) \|_1 \right].
$$

Multi-modality: We aim at having multi-modal outputs through the randomly sampled noise $z$ injected in $M$. However, to further encourage $M$ to produce diverse outputs, we employ the diversity sensitive loss [6], [40]:

$$
\mathcal{L}_{\text{ds}} = \mathbb{E}_{w,d(z_1, z_2) \sim \mathcal{N}(0, 1)} \left[ \| G(M(w, z_1, d) - G(M(w, z_2, d)) \|_1 \right],
$$

where $M$ is conditioned on two random noise vectors $z_1$ and $z_2$. We maximize $\mathcal{L}_{\text{ds}}$ to have generated images that are as different as possible from each other. To stabilize the learning process, we linearly decay the weight of the loss to zero during the training.

Overall Objective: The full objective functions can be summarized as follow:

$$
\min_{M} \max_{D} = \lambda_{\text{iff}} \mathcal{L}_{\text{iff}} + \lambda_{\text{cls}} \mathcal{L}_{\text{cls}} + \lambda_{\text{cont}} \mathcal{L}_{\text{cont}} + \lambda_{\text{nb}} \mathcal{L}_{\text{nb}} + \lambda_{\text{cyc}} \mathcal{L}_{\text{cyc}} - \lambda_{\text{ds}} \mathcal{L}_{\text{ds}}
$$

where $\lambda_{\text{iff}}, \lambda_{\text{cls}}, \lambda_{\text{cont}}, \lambda_{\text{nb}}, \lambda_{\text{cyc}}$ and $\lambda_{\text{ds}}$ are the hyper-parameters for each loss term. We note that the minus term before $\lambda_{\text{ds}}$ allows to maximize (8).

B. Injecting the Domain and Multi-Modality

As shown in Fig. 2, $M$ is implemented through two MLP function $f_1$ and $f_2$ and an AdaLN layer between them, which
we explain in this Section. The non-linear $f_1$ function maps the concatenation of a domain vector and a random noise to a hidden latent variable $h = f_1(d \oplus z)$. Then, $h$ goes through AdaLN, which is based on the widely used Adaptive Instance Normalization (AdaIN) [30]. Several state of the art approaches in I2I translation use AdaIN to inject the style features into the generating process and thus transfer the desired style into the output image. AdaIN assumes that images with the same style have a common mean and variance. Thus, existing methods usually extract the channel-wise mean and variance from images. However, previous literature has shown that the channels in the $\mathcal{W}^+$ latent codes of StyleGAN are correlated. For example, Karras et al. [18] has shown that bottom style layers in $W$ (i.e., the first channels of $\mathcal{W}^+$) control high-level aspects such as pose, general hair style, face shape, and eyeglasses, middle-layers are about facial features, hair style, eyes open/closed, while higher layers are about color scheme and micro-structure.

Thus, we propose a simple modification to AdaIN by replacing Instance Normalization [32] with Layer Normalization:

$$
\text{AdaLN}(w, \gamma, \beta) = \frac{w - \mu(w)}{\sigma(w)} + \beta,
$$

where the $\mu(\cdot)$ and $\sigma(\cdot)$ are calculated by the Layer Normalization, $\gamma$ and $\beta$ are parameters generated by a linear mapping function with the input $h$. Layer Normalization [31] computes the normalization for each sample across all the channels, avoiding destroying valuable information in the latent code. The parameters in the AdaLN module are learnt during the training.

The output of AdaLN goes through a second MLP function $f_2$ to obtain the edited style code $w^* = f_2(\text{AdaLN}(w, \gamma, \beta))$.

IV. EXPERIMENTS

We focus the experiments of the proposed ISF-GAN on StyleGAN v1 [18] and StyleGAN v2 [19], the two most prominent state-of-the-art unconditional GANs. However, we note that our methodology can be applied to several GANs such as PG-GAN [45].

**Baselines:** We compare our proposal with two state-of-the-art methods employing pre-trained and fixed unconditional GAN to manipulate faces: InterFaceGAN [23] and StyleFlow [13]. Moreover, we compare the ISF-GAN to StarGAN v2 [6] and its smooth extension [25] on gender translation, which are two state-of-the-art methods for MIMUTs. We use the officially released code for all the baselines in all these comparisons.

**Datasets:** InterFaceGAN [23] and StyleFlow [13] were trained and tested on different conditions. To have a fair evaluation between the proposed method and the baselines, we collect the following datasets:

- **Set1:** we randomly sample 90 K latent codes and collect the corresponding images through StyleGAN v1 [18] pre-trained on FFHQ dataset [18]. Then, we use an off-the-shelf classifier [18] to label each latent code with the corresponding semantic attributes. We randomly split the dataset into 80 K and 10 K samples for the training and testing sets;
- **Set2:** we collect the dataset released by StyleFlow [13], where 10 K training images and 1 K testing images are provided with annotated attributes and latent codes for StyleGAN v2 [19] pretrained on FFHQ dataset [18].

In each dataset, we model four key facial attributes commonly-used by InterFaceGAN [23] and StyleFlow [13] for analysis, including gender, smile (expression), age, and eyeglasses. We note that it is easily possible to plug in more semantic attributes as long as an attribute classifier is available.

**Evaluation metrics:** We evaluate both the visual quality and the diversity of generated images using Fréchet Inception Distance (FID) [54], the Learned Perceptual Image Patch Similarity (LPIPS) [53] and the Accuracy, evaluated through an off-the-shelf classifier provided by [18]. The details on evaluation metrics and protocols are further described in the Supplementary Materials (SI).

Moreover, we propose to use the state-of-the-art face recognition method ArcFace [55] to evaluate the content preservation in MIMUTs. We define a new metric called Face Recognition Similarity (FRS) that estimates the similarity between features extracted from two facial images. More formally:

$$
\text{FRS} = E_{s, z, d} [\langle \omega(G(s)), \omega(G(M(s, z, d))) \rangle]
$$

where $\omega(\cdot)$ refers to the pre-trained ArcFace network, $\langle \cdot, \cdot \rangle$ refers to cosine similarity between two input vectors. The extracted face regions are resized to $112 \times 112$ before being fed into $\omega$. An high FRS score shows the two faces have more similar identities, which indicates that a model better preserves the identities of faces during image manipulations.

**Training Details:** ISF-GAN is trained for 40 K iterations with batch size 4. The training time takes about 16 hours on 4 Tesla V100 GPUs with our implementation in PyTorch [56]. Compared to the state-of-the-art MIMUT model StarGAN v2 [6], the training time is reduced by half while having a much higher resolution (i.e., from $256 \times 256$ to $1024 \times 1024$). We set $\lambda_{r} = 1$, $\lambda_{cls} = 1$, $\lambda_{cont} = 1$, $\lambda_{adv} = 0.1$, $\lambda_{cycle} = 1$ and $\lambda_{dis} = 2.0$. We adopt the non-saturating adversarial loss [51] with R1 regularization [57] using $\gamma=1$. We use the Adam [58] optimizer with $\beta_1=0$ and $\beta_2=0.99$. The learning rates for $M$ and $D$ are set to $10^{-5}$.

A. Latent Codes Manipulation

We start by evaluating the performance of our framework on manipulating StyleGAN latent codes. Given a randomly sampled StyleGAN latent code $w$ with its corresponding attribute vector $d$, we choose a desire attribute vector $d^*$ and use $M$ to estimate a target $w^*$. Intuitively, an ideal model should well manipulate the original latent code $w$ to make the generated image $\alpha^* = G(w^*)$ with correct semantics. Fig. 3 suggests that the proposed ISF-GAN correctly learns the different semantics in StyleGAN v1 latent codes and allows to generate realistic translations. Compared to InterFaceGAN, our synthesized faces preserve the identity of the input image better, while correctly lying on the desired semantics. Similarly, as shown in Fig. 4, the proposed method not only translate the input images with correct visual semantics but better preserves the face identity also in StyleGAN v2.
Fig. 3. Visual comparisons between InterFaceGAN [23] and ISF-GAN on various attribute manipulations tested on the Set$_1$.

As shown in Table I, the overall quantitative evaluation verifies the superiority of the proposed method. The proposed ISF-GAN achieves better performance on FID and LPIPS, which indicates ISF-GAN synthesizes more realistic images and enables the generation of diverse images. This observation is consistent with the visual comparisons presented in Fig. 3 and Fig. 4. Table I shows that InterFaceGAN scores 0.00 in the LPIPS as the model is deterministic and does not allow diverse translations for the same input code. The only way to have diversity is to generate style latent codes in very close range to the translated latent code. This way scores 0.06 and 0.05 for Set$_1$ and Set$_2$, which is a result comparable to StyleFlow. However, changing the style codes in this way might affect the entire content of the generated image, including the identity of the person. Thus, it is not trivial to have multi-modal translations with InterFaceGAN.

We note that ISF-GAN can modify more than one attribute at a time, as shown in Fig. 5 while InterFaceGAN [23], [24] relies on multiple learned hyper-planes and requires users to translate multiple attributes in multiple steps (e.g., female $\rightarrow$ male $\rightarrow$ smile). StyleFlow [13] can manipulate multiple attributes at a time, but achieves lower accuracy than our model on the domain label used as a target of the translation (see mAcc in Table I).

We also note that we could not test Style-Flow on Set$_1$ as the authors did not release the model for StyleGAN v1. Thus, we could not have a fair comparison with them.

Content preservation: Quantitatively evaluating content preservation is very challenging. However, in face translation, the content is mainly related to the identity of people. Thus, we here evaluate whether the identity is preserved during the manipulation tasks. Table I shows that FRS increases by 121% in InterFaceGAN with Set$_1$. In Set$_2$, improvements are only slight perhaps due to the smoother style latent space of StyleGANv2. Fig. 6 shows that the identity preservation is also stable when we use randomly sample different noise codes $z$.

Latent space separation: The FID, LPIPS and FRS results do not come at the expense of the semantic correctness of generated images. We calculate the classification accuracy of the
TABLE I
QUANTITATIVE COMPARISONS ON IMAGE QUALITY, DIVERSITY, CONTENT PRESERVATION AND ACCURACY OF GENERATED IMAGES BASED ON PRE-TRAINED STYLEGAN V1 [18]/V2 [19]. THE PROPOSED ISF-GAN OUTPERFORMS ALL STATE-OF-THE-ART METHODS

| Model             | Set1 | Set2 |
|-------------------|------|------|
|                   | FID↓ | LPIPS↑ | FRS↑ | mAcc(%)↑ | FID↓ | LPIPS↑ | FRS↑ | mAcc(%)↑ |
| StyleFlow [13]    | -    | -     | -    | -       | 50.42 | 0.04  | 0.62 | 72.23    |
| InterFaceGAN [24] | 50.89| 0.00  | 0.32 | 94.17   | 29.85 | 0.00  | 0.66 | 78.95    |
| Ours              | 29.44| 0.16  | 0.75 | 95.45   | 23.93 | 0.22  | 0.66 | 96.52    |

TABLE II
COMPARISONS ON THE SMOOTHNESS OF SEMANTIC INTERPOLATIONS ON GENDER TRANSLATION

| Method             | Dataset | PPL↓ | PIR↓ |
|--------------------|---------|------|------|
| InterFaceGAN [24]  | Set1    | 28.76| 0.06 |
| Ours               |         | 10.23| 0.01 |
| StyleFlow [13]     |         | 63.06| 0.11 |
| InterFaceGAN [24]  | Set2    | 60.47| 0.07 |
| Ours               |         | 62.06| 0.07 |

Fig. 6. Multi-modal results obtained randomly sampling multiple z and collect diverse synthesized images.

B. Semantic Interpolation

Evaluating semantic interpolations is important for two main reasons. On the one hand, semantic interpolation allows to generate a smooth flow between two semantics (e.g., males and females), enabling users to generate intermediate images (e.g., non-binary gender or people transitioning gender) that are rarely present in existing datasets. On the other hand, interpolating latent codes allows qualitatively and quantitatively evaluating the smoothness of the latent space.

To semantically interpolate in the StyleGAN space, we first randomly sample a style s, then we manipulate its semantic with \( s^* = \mathcal{M}(s, z, d) \), where \( d \) changes exactly one attribute from the attribute vector of \( s \). Then, we linearly interpolate \( T \) equi-distant points along the interpolation line between \( s \) and \( s^* \). Finally, we generate the images in the path: \( \{G(s_0), G(s_1), \ldots, G(s_T)\} \), where \( s_0 = s \) and \( s_T = G(s^*) \).

SI Fig. 15 shows multiple semantic interpolations where we change the age, gender, expression and eyeglasses of some randomly sampled StyleGAN latent codes. We observe that the interpolations are very smooth, having almost imperceptible perceptual differences between neighbouring images.

To quantitatively evaluate the smoothness of interpolated images we rely on the Perceptual Path Length (PPL) [18] and a metric we propose: the Perceptual Interpolation Range (PIR). PPL measures the perceptual variation between pairs of images generated under small perturbations \( \epsilon \) in the latent space:

\[
PPL = \mathbb{E}_{s, s' \sim S, t \sim U(0,1)} \left[ \frac{1}{t^2} \Phi(\text{linterp}(s, s^*, t), \text{linterp}(s, s^*, t+\epsilon)) \right]
\]

where \( \Phi(s_i, s_j) = d(G(s_i), G(s_j)) \), and \( d(\cdot, \cdot) \) refers to the L2 distance between VGG [59] features.

PIR instead measures the semantic smoothness along the entire interpolation path between two latent codes. Specifically, PIR computes the range between the maximum and the minimum perceptual distance between two consecutive style codes along the interpolation path as \( \max_{t=1}^{T} \phi(s_{t-1}, s_t) - \min_{t=1}^{T} \phi(s_{t-1}, s_t) \). Very smooth interpolations between \( s_0 \) and \( s_T \) are expected to have \( \max_{t=1}^{T} \phi(s_{t-1}, s_t) \approx \min_{t=1}^{T} \phi(s_{t-1}, s_t) \), thus resulting in \( PIR \approx 0 \). Interpolations with abrupt changes would have \( \max_{t=1}^{T} \phi(s_{t-1}, s_t) \gg \min_{t=1}^{T} \phi(s_{t-1}, s_t) \), resulting in a high PIR. Since the perceptual distance along the interpolation path might be different depending on the sampled \( s_0 \) and \( s_T \) we normalize the computed

generated images by using off-the-shelf classifiers, in which the test set is composed of 1,000 positive samples and 1,000 negative samples for each manipulated attribute. In Table I, our proposed method achieves better performance in both the Set1 and Set2. We also compare the semantic accuracy of our method with the baselines in SI Table I. Our proposed method achieves the best performances on almost all of the four attributes. It indicates that ISF-GAN correctly disentangles the different semantics contained in the StyleGAN latent code better than the start-of-the-art methods, thus allowing to synthesize images into the desired semantic specified by attribute vector \( d \).
Fig. 7. Comparisons on gender interpolations between InterFaceGAN [23] and our proposed method tested on Set1. Compared to InterFaceGAN, our method preserves better the face identity along the interpolations.

range by $\phi(s_1, s_T)$. Thus:

$$PIR = E_{s_1, s_T \sim S} \left[ \max_{t=1}^{T} \phi(s_{t-1}, s_t) - \min_{t=1}^{T} \phi(s_{t-1}, s_t) \right]$$

$$\phi(s_1, s_T) + \epsilon$$

(13)

where $\epsilon$ is used for numerical stability. PIR uses the LPIPS [53] as the perceptual distance $d$, as LPIPS has been shown to be well aligned with human perceptual similarity.

Table II shows the quantitative results of the tested models. Both the PPL and PIR results show that the latent codes of our model are less entangled and that the perceptual path is substantially shorter than InterFaceGAN in Set1. As shown in Fig. 7, our proposed method not only generates smoother interpolations but also preserves better the face identity. Similarly, although the PPL results are comparable between StyleFlow, InterFaceGAN and the proposed ISF-GAN in Set2, PIR results suggest that there are less drastic changes between images generated along the interpolation path in ISF-GAN.

C. Comparisons With Traditional MMUIT Models

We evaluate ISF-GAN with the state-of-the-art model of MMUITs, namely StarGAN v2 [6]. We test all the models on gender translation, following the main results of StarGAN v2.

ISF-GAN achieves better performance compared to StarGAN v2 (20.67 vs 71.20 on FID score). Fig. 8 shows that ISF-GAN not only generates more realistic images with correct semantics but also preserves better face identity and background. We also compare our model with SmoothLatent [25], which improves StarGAN v2 to have smooth inter-domain interpolations. ISF-GAN achieves better performance compared to SmoothLatent (0.050 vs 0.062 on PIR score). SI Fig. 13 shows that the inter-domain interpolations are smoother in both visual semantics and background.

We note that training ISF-GAN requires fewer resources than training StarGAN v2 and its extension.

D. Manipulating Real Face Images

Fig. 9 shows an example of manipulation of real images. First, we map the source images into the StyleGAN latent space through Abdal et al. [11], in which they search the latent code that best approximates the given image. Then, we use our model to translate images into multiple semantics. The results show that ISF-GAN can achieve high-quality results with such inverted latent codes of real images. Moreover, it indicates a promising application that ISF-GAN is used in conjunction with an off-the-shelf method to do GAN inversion in StyleGAN (either through an encoder or an optimization method) for unsupervised image-to-image translation. SI Fig. 11 shows some additional examples on CelebA-HQ.

E. Beyond Face Translations

To verify the generalization of the proposed method, we apply it to a two-domain image translation (e.g., Cat ↔ Dog) task. We collect a pre-trained StyleGAN v2 on the AFHQ dataset [6] (using only “cat” and “dog” images). The model is trained with
images at lower resolution (256×256 resolution), batch size 8 and 1e5 iterations on 4 Tesla V100 GPUs. SI Fig. 12 shows several randomly sampled generated images based on the pre-trained StyleGAN v2.

Similar to the protocol described in Section IV, we randomly sample 11 K latent codes (10 K for training and 1 K for testing) in the \( W^+ \) space and collect the corresponding images through the pre-trained GAN. Then, we also train a two-class classifier (i.e., a standard ResNet-50 [60]) on the AFHQ dataset to automatically distinguish the cat and dog faces. Finally, we train the proposed ISF-GAN on such pre-trained GAN and aim to do Cat ↔ Dog translation. As shown in Fig. 10, ISF-GAN performs well on this task, which indicates the excellent generalization ability of the proposed method.

**F. Ablation Study**

We here study the importance of each component of our proposed method. As shown in Table III, the FID, LPIPS, FRS and PIR values for all the configurations where we removed or replaced components of our proposal Table III-A. We repeated the experiments five times to compute the standard deviation of each metric.

First, we observe that FID and FRS significantly decrease by removing the neighbouring constraints \( L_{nb} \), but the diversity in the generated images (see Table III-B) is evidently improved. This result suggests that constraining the latent codes to be similar in features to neighbouring latent codes helps disentangle the attributes from the attribute-invariant part of the image, thus improving the content preservation and quality of synthesized images.

Second, when we remove \( L_{cont} \), which encourages manipulated images to be perceptually similar to the original image, the LPIPS also improves but FID and FRS decrease. It shows a similar tendency as Table III-B. However, \( L_{cont} \) has a more significant effect on the interpolation smoothness compared to \( L_{nb} \). This result suggests that perceptual loss is significant for better image quality and content preservation. From qualitative results, we see that \( L_{cont} \) helps the ISF-GAN better preserve some distinct face characteristics (e.g., the chin shape). The network without \( L_{cont} \) generates more diverse images because it also changes the identity of people.

We also ablate the proposed AdaLN by replacing it with AdaIN. Table III-D shows that all metrics get worse. This suggests that StyleGAN latent codes do not have a channel-wise style distribution and that a channel-wise normalization destroys valuable information in it. Thus, the experiment verifies that the proposed AdaLN is more suitable for manipulating the latent codes of StyleGAN.

Finally, we ablate the value of \( \lambda_{ds} \), which controls how much the model should focus on diversity. As expected, Table IV shows that the bigger is the value, the higher is the diversity of generated images. However, we also see that the changes on FID, FRS and PIR seem to be negatively correlated with the changes on LPIPS. This result is explained by the image quality and diversity trade-off previously discussed in [25]. It is easier for the model to collapse to images very similar to the target domain, thus obtaining a lower FID and LPIPS, than creating...
very diverse images that sometimes contain features (e.g. a lot of different hairstyles) that are not present in the source domain images.

In our paper, we chose to have a higher FID but having higher LPIPS. However, we note that our improvements in FID from the state of the art models are still substantial, as depicted in Table I.

V. CONCLUSION

In this paper, we propose an Implicit Style Function that enables Multi-modal and Multi-domain Unsupervised Image-to-Image Translations at high resolution without training a generator from scratch. Using pre-trained and fixed large-scale generators, we show that it is possible to manipulate and translate images with results that have higher quality, diversity, consistency and smoothness than state-of-the-art approaches, even if they are trained specifically for the MMUIT task.

ACKNOWLEDGMENT

This work was done during internship at Tencent AI Lab, Shenzhen, China.

REFERENCES

[1] C. Ledig et al., “Photo-realistic single image super-resolution using a generative adversarial network,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 4681–4690.
[2] S. Menon, A. Damian, S. Hu, N. Ravi, and C. Rudin, “Pulse: Self-supervised photo upsampling via latent space exploration of generative models,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2020, pp. 2437–2445.
[3] D. Lee et al., “Context-aware synthesis and placement of object instances,” Neural Inf. Process. Syst., vol. 31, 2018.
[4] Y. Jo and J. Park, “SC-FEGAN: Face editing generative adversarial network with user’s sketch and color,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2019, pp. 1745–1753.
[5] Y. Choi et al., “StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 8789–8797.
[6] Y. Choi, Y. Uh, J. Yoo, and J.-W. Ha, “StarGAN v2: Diverse image synthesis for multiple domains,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2020, pp. 8188–8197.
[7] B. Li, X. Qi, T. Lukasieicz, and P. Torr, “Controllable text-to-image generation,” Adv. Neural Inf. Process. Syst., vol. 32, 2019.
[8] Y. Liu et al., “Gmm-unit: Unsupervised multi-domain and multi-modal image-to-image translation via attribute gaussian mixture modeling,” 2020, arXiv:2003.06768.
[9] A. Sirorhon, S. Lathuilière, S. Tulyakov, E. Ricci, and N. Sebe, “First order motion model for image animation,” Neural Inf. Process. Syst., vol. 32, 2019.
[10] S. Tulyakov, M.-Y. Liu, X. Yang, and J. Kautz, “MoCoGAN: Decomposing motion and content for video generation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 1526–1535.
[11] R. Abdal, Y. Qin, and P. Wonka, “Image2StyleGAN: How to embed images into the StyleGAN latent space?” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2019, pp. 4342–4441.
[12] R. Abdal, Y. Qin, and P. Wonka, “Image2StyleGAN++: How to edit the embedded images?,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2020, pp. 8296–8305.
[13] R. Abdal, P. Zhu, N. Mitra, and P. Wonka, “StyleFlow: Attribute-conditioned exploration of StyleGAN-generated images using conditional continuous normalizing flows,” ACM Trans. Graph., vol. 40, no. 3, 2021, doi: 10.1145/3447684.
[14] R. Abdal, P. Zhu, N. Mitra, and P. Wonka, “Labels4Free: Unsupervised segmentation using StyleGAN,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), 2021, pp. 13970–13979.
[15] X. Pan, B. Dai, Z. Liu, C. C. Loy, and P. Luo, “Do 2D GANs know 3D shape? unsupervised 3D shape reconstruction from 2D image GANs,” in Proc. Int. Conf. Learn. Representations, 2021.
[16] Y. Tian et al., “A good image generator is what you need for high-resolution video synthesis,” in Proc. Int. Conf. Learn. Representations, 2021.
[17] Y. Zhang et al., “Image GANs meet differentiable rendering for inverse graphics and interpretable 3D neural rendering,” in Proc. Int. Conf. Learn. Representations, 2021.
[18] T. Karras, S. Laine, and T. Aila, “A style-based generator architecture for generative adversarial networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2019, pp. 4401–4410.
[19] T. Karras, “Analyzing and improving the image quality of StyleGAN,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2020, pp. 8110–8119.
[20] E. Collins, R. Bala, B. Price, and S. Susstrunk, “Editing in style: Uncovering the local semantics of GANs,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2020, pp. 5771–5780.
[21] E. Härkönen, A. Hertzmann, J. Lehtinen, and S. Paris, “GANSPACE: Discovering interpretable GAN controls,” Neural Inf. Process. Syst., vol. 33, pp. 9841–9850, 2020.
[22] Z. Wu, D. Lischinski, and E. Shechtman, “Style-space analysis: Dsitangled controls for StyleGAN image generation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2021, pp. 12863–12872.
[23] Y. Shen, J. Gu, X. Tang, and B. Zhou, “Interpreting the latent space of GANs for semantic face editing,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2020, pp. 9243–9252.
[24] Y. Shen, C. Yang, X. Tang, and B. Zhou, “InterFaceGAN: Interpreting the disentangled face representation learned by GANs,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 44, no. 4, pp. 2004–2018, Apr. 2022.
[25] Y. Liu et al., “Smoothing the disentangled latent style space for unsupervised image-to-image translation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2021, pp. 10785–10794.
[26] X. Huang, M.-Y. Liu, S. Belongie, and J. Kautz, “Multimodal unsupervised image-to-image translation,” in Proc. IEEE Eur. Conf. Comput. Vis., 2018, pp. 172–189.
[27] H.-Y. Lee et al., “DRIT++: Diverse image-to-image translation via disentangled representations,” Int. J. Comput. Vis., vol. 128, no. 10, pp. 2402–2417, 2020.
[28] Y.-C. Chen, X. Xu, Z. Tian, and J. Jia, “Homomorphic latent space interpolation for unpaired image-to-image translation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2019, pp. 2408–2416.
[29] F. Richardson et al., “Encoding in style: A StyleGAN encoder for image-to-image translation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2021, pp. 2287–2296.
[30] X. Huang and S. Belongie, “Arbitrary style transfer in real-time with adaptive instance normalization,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2017, pp. 1501–1510.
[31] L. Ba, J. R. Kiros, and G. E. Hinton, “Layer normalization,” 2016, arXiv:1607.06450.
[32] D. Ulyanov and V. Lempitsky, “Instance normalization: The missing ingredient for fast stylization,” 2016, arXiv:1607.08022.
[33] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, “Image-to-image translation with conditional adversarial networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 1125–1134.
[34] J.-Y. Zhu et al., “Toward multimodal image-to-image translation,” Neural Inf. Process. Syst., vol. 30, 2017.
[35] A. Sirorhin, E. Sangineto, S. Lathuilière, and N. Sebe, “Deformable GANs for pose-based human image generation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 3408–3416.
[36] J. Hoffman et al., “CyCADA: Cycle-consistent adversarial domain adaptation,” in Proc. Int. Conf. Mach. Learn., 2018, pp. 1989–1998.
[37] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2017, pp. 2223–2232.
[38] H.-Y. Lee, H.-Y. Tseng, J.-B. Huang, M. Singh, and M.-H. Yang, “Diverse image-to-image translation via disentangled representations,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2018, pp. 35–51.
[39] A. Pumarola, A. Agudo, A. M. Martínez, A. Sanfeliu, and F. Moreno-Noguer, “Ganimation: Anatomically-aware facial animation from a single image,” in Proc. IEEE Eur. Conf. Comput. Vis., 2018, pp. 818–833.
[40] Q. Mao, H.-Y. Lee, H.-Y. Tseng, S. Ma, and M.-H. Yang, ‘‘Mode seeking generative adversarial networks for diverse image synthesis,’’ in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2019, pp. 1429–1437.
Yahui Liu received the B.S. and M.S. degrees from Wuhan University, Wuhan, China, in 2015 and 2018, respectively. He is currently working toward the Ph.D. degree with the Department of Information Engineering and Computer Science, University of Trento, Trento, Italy. His main research interests include machine learning and computer vision, including unsupervised learning, and image domain translation.

Yajing Chen received the B.S. degree from the Department of Computer Science and Technology, Shanghai Jiao Tong University, Shanghai, China, in 2015. She is currently a Researcher with Tencent AI Lab. Her research interests include 3D face reconstruction and image synthesis.

Linchao Bao received the M.S. degree in pattern recognition and intelligent systems from the Huazhong University of Science and Technology, Wuhan, China, and the Ph.D. degree in computer science from the City University of Hong Kong, Hong Kong, in 2015. From November 2013 to August 2014, he was a Research Intern with Adobe Research. From January 2015 to June 2016, he was an Algorithm Engineer with DJI. He is currently a Principal Research Scientist with Tencent AI Lab. His research interests include computer vision and graphics.

Nicu Sebe (Senior Member, IEEE) is currently a Professor with the University of Trento, Trento, Italy, leading the research in the areas of multimedia information retrieval and human behavior understanding. He was the General Co-Chair of the IEEE FG 2008 and the Association for Computing Machinery (ACM) Multimedia 2013. He was a Program Chair of the ACM Multimedia in 2011 and 2007, European Conference on Computer Vision (ECCV) 2016, International Conference on Computer Vision 2017, and International Conference on Pattern Recognition (ICPR) 2020. He is the General Chair of ACM Multimedia 2022 and a Program Chair of ECCV 2024. He is a Fellow of the International Association of Pattern Recognition.

Bruno Lepri received the Ph.D. degree in computer science from the University of Trento, Trento, Italy, in 2009. He currently leads the Mobile and Social Computing Laboratory, Fondazione Bruno Kessler (FBK), Trento, Italy. From 2010 to 2013, he was a Joint Postdoctoral Fellow with the MIT Media Laboratory, Boston, MA, USA, and FBK. His research interests include computational social science, human behavior understanding, and machine learning. He was the recipient of the James Chen Annual Award for the Best UMLAI Paper, and Best Paper Award from ACM Ubicomp in 2014. He was the Area Chair of ACM Multimedia 2020 and 2021.

Marco De Nadai received the master’s and Ph.D. degrees in computer science from the University of Trento, Trento, Italy. He is currently a Research Scientist with Fondazione Bruno Kessler, Trento, Italy. His expertise lies in human behavioural understanding through the analysis of multi-modal data, computer vision, and unsupervised learning. He collaborated with numerous international institutions, including MIT, MIT Media Lab, Nokia Bell Labs, Vodafone Group, and Data-pop Alliance. He was awarded by the University of Trento for the Best Master’s and Best Ph.D. Student.