# Image Generation using Continuous Filter Atoms

**Anonymous Author(s)**
Affiliation
Address
email

## Abstract

In this paper, we model the subspace of convolution filters with a neural ordinary differential equation (ODE) to enable gradual changes in generated images. Decomposing convolutional filters over a set of filters atoms allows efficiently modeling and sampling from a subspace of high-dimensional filters.By further modeling filter atoms with a neural ODE, we show both empirically and theoretically, that such introduced continuity can be propagated to the generated images, and thus achieves smoothly evolved image generation. We validate the proposed framework of image generation with continuous filter atoms on various experiments, including image to image translation and image generation conditioned on continuous labels. Without auxiliary network components and heavy supervisions, the proposed continuous filter atoms allow us to easily manipulate the gradual change of generated images by controlling the integration interval of neural ordinary differential equation. This research sheds the light on using the subspace of network parameters to navigate the diverse appearance of image generation.

## 1 Introduction

Conditional image generation has been widely studied in recent years due to their wide applications including image segmentation, style transfer, image inpainting, image super-resolution, image registration, and image synthesis [26]. Despite extensive research and application in these fields, limited progress has been made on conditional image generation using continuous or closely spaced labels due to difficulties pointed out in [7], e.g., the absence of real images for some labels. Moreover, it remains challenging to encourage generation diversity, especially without heavy supervision, while maintaining output fidelity. Previously, generation diversity is mainly encouraged using explicit regularization terms to convey diversity with additional latent codes [21, 29]. However, explicit regularizations inevitably introduce additional hyperparameters, and sub-optimal hyperparameters can often cause either poor diversity or noticeable sacrifices in terms of generation quality and the correspondence to the input condition.

On the other hand, achieving generation diversity by modeling the deep network parameter space has attracted limited attention [39]. This direction is mainly restricted by the prohibitive cost of both modeling and sampling of the very high-dimensional space of convolutional filters in modern image generative networks. Motivated by the observation that a convolutional filter can be well approximated by a linear combination of low-dimensional filter atoms, BasisGAN [39] shows in image generation that, the high-dimensional filter space in each layer can be well approximated by a low-dimensional filter subspace, which significantly reduces the cost of both parameter modeling and sampling. In this way, each sampling of the modeled parameter subspace results in one deterministic transformation from the input condition to the desirable target domain, and the diverse outputs are achieved by sampling multiple times. More importantly, it is empirically shown in [39] that without any explicit regularizations, such parameter subspace modeling can work as a plug-and-play module to convert a deterministic conditional image generation network to produce diverse images with

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

appealing stochasiticity, without any auxiliary network components or regularization terms to add hyperparameters requiring heavy tuning.

However, we observe that the filter subspace modeling in [39] suffers from several major limitations: First, this method is sensitive to network configurations, and mode collapse, which results in a point estimation to the subspace of parameter, can often be observed. Furthermore, this method is incapable of modeling a continuous space, so that gradual changes to the generated parameters cannot be obtained by simply interpolating samples; and such discontinuity will be further propagated to the output, e.g. crashed and unnatural generated images, and lead to unstable training [1].

In this paper, we adopt a subspace view to convolutional filters by performing atom-coefficients decomposition, as in [39]. Then we further model the filter subspace using neural ordinary differential equation (ODE), so that we are able to produce a continuous series of filter samples at arbitrarily fine resolution. The low-dimensional filter subspace allows extremely efficient learning and modeling using ODE. We show both empirically and theoretically that, continuous transition in the filter subspace can be propagated naturally to the generated images, and thus produce visually appealing images with gradually varying appearance. More importantly, we show that the continuity introduced to the filter subspace and the induced generation smoothness allow continuous manipulations of generated appearance with only discrete samples as supervisions, which is a task that is used to be considered intractable by training with standard empirical risk minimization [7]. These appealing properties of the proposed method enable various significant applications through only standard training, without requiring any auxiliary network components, additional regularization terms, or heavy supervisions.

We perform various experiments on conditional image generation, with conditions being in the form of images, labels, or both. We list here several example applications that are enabled by the proposed image generation using continuous filter atoms from an ODE atom generator,

- Continuous image synthesis that covers a wide range of gradually varying appearance with high fidelity and accurate correspondence to the input condition.
- Sequential image synthesis with explicitly specified starting and ending points to allow flexible appearance manipulation without heavy supervisions.
- Interpolation of generated image appearance at arbitrarily fine resolution.
- An effective approach for generating images conditioned on continuous labels.

## 2 Related Work

**Conditional image generation.** Image generation is widely studied in the machine learning community, from restricted Boltzmann machines [34] to variational autoencoders [17]; in particular variants with conditions [25, 35, 36] show promising results. Empowered by GANs [10], conditional generative adversarial networks (cGANs) [15, 28, 33, 37, 41, 46, 21, 39] have demonstrated superior performance on image generation with the restriction of condition. Recently, conditional image generation with condition on continuous labels has attract attention, and new training schemes [7] are proposed to improve continuous conditional image generation. From the perspective of the dynamic of parameter subspace, our method provides a simple yet effective solution.

**Continuous image translation.** Many works have discussed improving the diversity of generated images in image-to-image translation tasks [39, 29, 5, 21, 19, 38, 13, 43, 4]. However, continuous image translation that aims to generate intermediate images between the source and the target images is not yet widely studied. Some previous works [44, 42, 22, 40, 9] have manipulated the attribute vectors either by interpolation or linear transformations to continuously control the latent space. [14, 20] employ multiple generative networks in the middle of the path way from source to target image to generate intermediate results. [5, 31] disentangle the latent space as content and style codes and interpolated the styles, in order to produce intermediate samples. However, all the aforementioned methods rather need to import auxiliary neural networks or carry extra burdens on model training by introducing additional regularization or supervisions. Moreover, synthesizing intermediate images without relying on interpolation of either domain labels, or disentangled representations, which can discourage the stochasticity of generated images, is still considered a big challenge.

# 3 Continuous Filter Atoms

In this section, we start with a brief review on neural ODEs and atom-coefficient filter decomposition, which serve as important ingredients in the proposed image generation approach. We then present in detail that, with the effectiveness being theoretically justified, how the proposed continuous atom generation using neural ODEs achieves smooth image generation.

## 3.1 Neural ODEs

Neural ordinary differential equations (ODEs) are introduced in [3] as an approach to interpret the neural network as a system of ODEs, where each ODE represents the underlying dynamics of the hidden elements. Especially, neural ODEs are frequently adopted to enforce latent representations to follow continuous dynamics. [16, 27] propose to use neural ODEs to smoothly connect the latent space of video frames. [32] adopts ODE-RNNs to model time-continuous hidden dynamics to handle irregularly sampled time-series data.

Neural ODEs model a latent state $\mathbf{z}(t_s)$ as $\mathbf{z}(t_s) = \mathbf{z}(t_0) + \int_{t_0}^{t_s} f(\mathbf{z}(t), t; \boldsymbol{\theta})\, dt$, where $\frac{d\mathbf{z}(t)}{dt} = f(\mathbf{z}(t), t; \boldsymbol{\theta})$ is modeled as a neural network parametrized by $\boldsymbol{\theta}$. Then, the latent state at an arbitrary point $t_s$ can be obtained as

$$\mathbf{z}(t_s) = \text{ODESolve}(\mathbf{z}(t_0), f, (t_0, t_s), \theta), \tag{1}$$

where $t_0$ and $t_s$ denote the start and the end of the integral interval, respectively. The simplest method for approximating the solutions of the ordinary differential equations with a given initial value is the Euler's method, which is a first-order integrator with a fixed step-size. In practice, however, families of Runge-Kutta methods, e.g., the Midpoint method, and fourth-order Runge-Kutta [18], are preferred due to their superior convergence and stabilities. Note that all the experiments in this paper follow the implementation in [3] for Runge-Kutta of fifth-order of Dormand-Prince-Shampine with adaptive step-size [8].

For any given integral interval $(t_i, t_j)$ where $i, j \in \{0...T\}$, ODESolver always outputs a unique solution for the integral of the continuous dynamics, as long as the ordinary equation $f$ is uniformly Lipschitz continuous in $\mathbf{z}$ and $t$ [6]. Parameterizing $f$ as a neural network equipped with Lipschitz continuous non-linear functions will meet the requirements. Thus, in our paper, modeling filter subspsace using neural ODEs is theoretically invertible and guarantees continuity. This guaranteed convertible property, together with additional continuity property that we will theoretically justify later, jointly build the foundation of the proposed smooth image generation with continuous filter atoms.

## 3.2 Convolutional Filter Decomposition

Directly modeling the space of high-dimensional convolutional filters in an image generation network is practically prohibitive in terms of both computation and parameter scale [39]. Inspired by the observation that a convolutional filter can be well-approximated as a linear combination of filter bases [30], as shown in [39], a subspace view to the convolutional filters can be adopted in image generation network by decomposing each convolutional filter as a linear combination of low-dimension filter atoms. Specifically, given a $l \times l$-sized convolutional filter $\mathbf{F} \in \mathbb{R}^{c \times c' \times l \times l}$ with $c'$ input and $c$ output channels, respectively, we can decompose the filter over a set of $m$ atoms $\mathbf{D}$ as $\mathbf{F} = \mathbf{AD}$, where $\mathbf{D} \in \mathbb{R}^{m \times l \times l}$, and $\mathbf{A} \in \mathbb{R}^{c \times c' \times m}$ are the filter subspace coefficients. As we will show later, by modeling the low-dimensional filter subspace with continuity using neural ordinary differential equations, we achieve efficient training and modeling of gradual changes in conditional image generations.

## 3.3 Continuous Atom Generation for Smooth Appearance Modeling

In this paper, we target for modeling gradual changes in generated images with respect to continuous input conditions. Instead of seeking new ways of training, we provide a unique view by modeling the transformation from the input conditions to the corresponding output with parameters instantiated from an underlying continuous space, which subsequently leads to generated images with gradual changes.

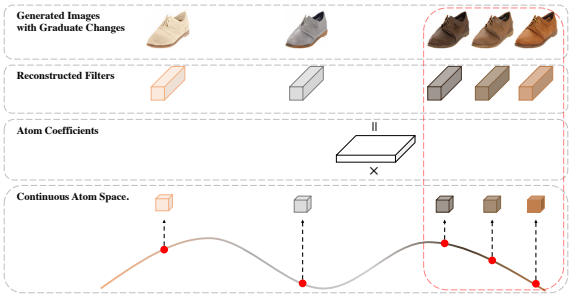


Figure 1: Illustration of the proposed image generation using continuous filter atoms. Note that only a single convolutional layer is shown here for easier viewing. We decompose each convolutional filter as $\mathbf{F} = \mathbf{AD}$, where the coefficients $\mathbf{A} \in \mathbb{R}^{c \times c' \times m}$ remain learned deterministically, and the filter atoms $\mathbf{D} \in \mathbb{R}^{m \times l \times l}$ are sampled from a continuous space using neural ODE. As shown in the red box, continuously sampled atoms can produce images with graduate appearance changes.

We adopt the aforementioned atom-coefficient decomposition, which permits low computational modeling of filter subspace as also validated in [39]. However, modeling a filter subspace with standard neural networks can still suffer from model collapse often caused by sub-optimal network configurations, and the continuity is also difficult to be guaranteed by standard training as raised in [7]. We therefore adopt a neural ODE based filter atom generator with the continuity property as discussed in Section 3.1, and show theoretically that such atom continuity can be propagated to the deep network outputs to achieve conditional image generation with gradual changes.

Specifically, we decompose $i$-th layer filters $\mathbf{F}^i$ into respective atoms $\mathbf{D}^i$ and coefficients $\mathbf{A}^i$, as $\mathbf{F}^i = \mathbf{A}^i\mathbf{D}^i$. Over $k$ layers of filters $\{\mathbf{F}^i\}_1^k$, we jointly model the corresponding atoms $\{\mathbf{D}^i\}_1^k$ by the atom generation. We assume all instantiations of filter atoms reside in a continuous space parameterized as $\boldsymbol{\mathcal{D}}(t)$ where $t$ is a time variable, i.e., the generated atoms at the time point $t$ can be expressed as $\boldsymbol{\mathcal{D}}(t)$. In the limit of a small step $\epsilon$, the local dynamic within the atom space can then be expressed as $\boldsymbol{\mathcal{D}}(t+\epsilon) = \boldsymbol{\mathcal{D}}(t) + \epsilon \cdot d\boldsymbol{\mathcal{D}}(t)/dt$. Therefore, given an initial state of atoms within this space denoted as $\boldsymbol{\mathcal{D}}(t_0)$, and modeling the derivative of $d\boldsymbol{\mathcal{D}}(t)/dt$ using a neural network $f(\cdot, \cdot; \theta)$, any sample within the atom space can be obtained by the neural network with inputs of $\boldsymbol{\mathcal{D}}(t)$ and $t$. The described dynamic leads naturally to a neural ordinary differential equation, where we have

$$\frac{d\boldsymbol{\mathcal{D}}(t)}{dt} = f(\boldsymbol{\mathcal{D}}(t), t; \theta). \tag{2}$$

Therefore, the instantiation at a time point $t_s, s > 0$, in this continuous dynamic can be obtained through a continuous atom generator $\mathcal{T}$ by

$$\boldsymbol{\mathcal{D}}(t_s) = \mathcal{T}(t_s; \boldsymbol{\mathcal{D}}(t_0), \theta) = \boldsymbol{\mathcal{D}}(t_0) + \int_{t_0}^{t_s} f(\boldsymbol{\mathcal{D}}(t), t; \theta) dt. \tag{3}$$

Note that the initial state $\boldsymbol{\mathcal{D}}(t_0)$ here are free parameters to be optimized in the training jointly with $\theta$ that parametrizes $f$. Starting from the initial state $\boldsymbol{\mathcal{D}}(t_0)$, an instantiation of filter atoms $\boldsymbol{\mathcal{D}}(t_s)$ is defined to be a unique solution to the ODE initial value problem at some condition $t_s$, by Picard's existence theorem. Consequently, as $t$ varies, different instantiations of atoms are generated from this modeled continuous filter subspace, and our corresponding atom generator equipped with ODESolver learns the underlying continuous dynamics of the filter subspace. Accordingly, the resulting subspace of convolutional filters becomes continuous.

Given the subspace of filters modeled as an underlying continuous dynamic through neural ODE, we prove the Lipschitz-type continuity transducted from atoms to output features in the $i$-th layer. For simplicity, let $c = c' = 1$, and the argument extends. Given an input image $X(u)$ ($u \in \mathcal{U}, \mathcal{U} \subset \mathbb{Z}^2$), define the local input norm $||x||_{2,N_u} := (\sum_{u' \in N_u} x(u-u')^2)^{1/2}$ and the convolution $\langle x, w \rangle_{N_u} := \sum_{u' \in N_u} x(u-u')w(u')$, where $N_u \subset \mathcal{U}$ is a local Euclidean grid centered at $u$. The $i$-th layer function $\mathcal{F}^i : X^{i-1} \to X^i$ in can be expressed as,

$$X^i(u) = \mathcal{F}^i(X^{i-1}, \mathbf{D}^i; \mathbf{A}^i) = \sigma(\sum_{k=1}^{m} \mathbf{A}_k^i \langle X^{i-1}, \mathbf{D}^{i,k} \rangle_{N_u} + b^i), \tag{4}$$

where $\mathbf{D}^{i,k}$ denotes the $k$-th atom in the $i$-th layer, $\mathbf{A}_k^i$ is the corresponded $k$-th coefficient, and $b^i$ denotes the bias at the $i$-th layer.

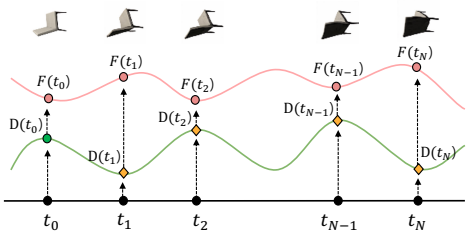


Figure 2: An example of generating rotating chairs with the proposed continuous filter atoms. Each $t$ in the figure represents a distinct angle along the roll axis, and is given as an explicit condition to the atom generation.

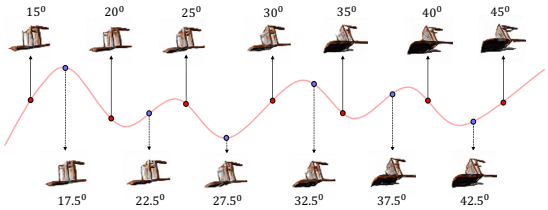


Figure 3: Visualization of interpolated samples in the rotating chair example. As the neural ODE-based atom generator learns a smooth trajectory of rotation, image generation at any desired degrees is now allowed. The images generated at the blue dots are interpolation results for unseen rotation angles.

**Theorem 1.** *Suppose* $\mathbf{D}_1^i$ *and* $\mathbf{D}_2^i$ *are two continuously generated atoms, (i.e.,* $\exists M > 0, ||\mathbf{D}_1^i - \mathbf{D}_2^i||_2 \leq M|t_1 - t_2|$*), and assume the activation function* $\sigma$ *is non-expansive which holds for ReLU, then* $\mathcal{F}^i$ *is continuous in* $\mathbf{D}^i$*,*

$$||X_1^i - X_2^i||_2 \leq (||\mathbf{A}^i||_2 \lambda)\sqrt{|\mathcal{U}|} \cdot ||(\mathbf{D}_1^i - \mathbf{D}_2^i)||_2, \quad \textit{with } \lambda = \sup_{u \in \mathcal{U}} ||X^{i-1}||_{2,N_u}, \tag{5}$$

in which $X_1^i = \mathcal{F}^i(X^{i-1}, \mathbf{D}_1^i; \mathbf{A}^i), X_2^i = \mathcal{F}^i(X^{i-1}, \mathbf{D}_2^i; \mathbf{A}^i)$ are outputs correspond to two atoms.

*Proof.* The proof is provided in the Appendix.

Under the rule of neural ODE, the dimensionality of $\boldsymbol{\mathcal{D}}(t)$, $t > 0$, cannot be varied, meaning the dimension of the input and the output of the ODESolver remain strictly consistent. This additional restriction makes modeling the space of full-size filters infeasible due to the computation and memory constraints. Meanwhile, by modeling the filter subspace instead, we significantly reduce the cost of learning the corresponding neural ODE. Typically, atoms across layer usually introduce a few hundred of parameters only, and significantly simplify the training and inference of atom generator $\mathcal{T}$.

Note that depending on the specific applications of the proposed continuous atom generation as we will show next, the ending point of the ODE integral can be determined by: 1) explicitly given to strictly control the behavior of continuous dynamics, or 2) stochastically sampled from a prior distribution, e.g., Uniform distribution or Gaussian distribution, to enhance the diversity of the generated images. Explicitly designating the integration interval is similar to feeding label information directly to a generator or discriminator in a well-known concatenation based way [23], or a projection based way [24]. Such explicit modeling reconstructs the sets of continuous atoms along the learned continuous dynamics, and loads each of them as decomposed element of the filter according to the circumstances. A specific example of the proposed method is presented in the Figure 2, where our filter atoms learn the hidden continuous dynamic of rotating chairs with respect to the rotation degrees. Our proposed method is much more straight-forward than the previous conditional GAN methods [23, 24], as label information can now be supplied with minimal preprocessing, simply in the forms of raw floating numbers preserving the continuity of labels, instead of being vectorized or one-hot encoded. Moreover, unlike [23] which can only take finite discretized conditions, such as categorical class information, continuous filter atoms do not hold any such restrictions. Therefore, by instantiating the subspace of convolutional filters with ODE parameters defined by the raw labels, our continuous filter atoms can be evaluated at any desired states,.e.g, intermediate points of raw labels. For better illustration, we show interpolation results of images at the unknown intermediate states in Figure 3, by sampling filter atoms along the rotation-continuous dynamic.

The stochastic modeling of integration interval enhances the diversity of generated images. Although randomness of generated images cannot be strongly enforced as in [39] because of the Neural ODE that governs the behavior of generator, a gradual transition between images can be clearly witnessed instead. We present the respective results in the Section 4.1.

**Conditional image generation with continuous atoms.** By simply plugging in the proposed continuous atom generator, we show next that variants of conditional generative adversarial models

(cGAN) [10, 23] with continuity can be easily obtained. And the training does not involve any auxiliary network components or additional regularizations.

***Conditional on images.*** Image generation conditional on images, also known as image-to-image translation [15], aims at generating a target image with high fidelity while maintaining correspondence w.r.t. an input image condition, and desirable diversity [39, 21, 29]. We introduce image generation conditional on input images with the output space modeled as a smooth sequence through continuous atom filters, thus sampling the continuous filter atoms allows generating images with graduate appearance changes without losing the correspondence to the conditions. Formally, representing the condition images as $\mathbf{x}$, which are usually samples from an empirical distribution $p(\mathbf{x})$, we aim at modeling a conditional distribution $p(\mathbf{y}|\mathbf{x})$. By plugging in the continuous atom generator $\mathcal{T}(t; \boldsymbol{\mathcal{D}}_0, \theta)$, the conditional distribution is learned to be modeled by generator $\mathbb{G}_{\phi,\theta,\boldsymbol{\mathcal{D}}_0}$, where $\phi$ contains all remaining parameters in the generator, e.g., the coefficients $\mathbf{A}$ in each decomposed convolutional layer. And the input to the discriminator $\mathbb{D}$ is now trained to evaluate the compatibility of a pair of images, one from the source domain $\mathbf{x} \sim p(\mathbf{x})$, the other one comes from either true target distribution $\mathbf{y} \sim p(\mathbf{y}|\mathbf{x})$, or the generated approximation $\mathbb{G}_{\phi,\theta,\boldsymbol{\mathcal{D}}_0}\big(\mathbf{x}; \mathcal{T}(t; \boldsymbol{\mathcal{D}}_0, \theta)\big)$.

The training can be realized as

$$\begin{aligned}\min_{\mathbb{G}} \max_{\mathbb{D}} \mathcal{L}(\mathbb{D}, \mathbb{G}) =& \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x}), \mathbf{y}\sim p(\mathbf{y}|\mathbf{x})}\big[\log\big(\mathbb{D}(\mathbf{x}, \mathbf{y})\big)\big] + \\ & \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x}), t\sim \text{Uniform}(t_0, t_{-1})}\Bigg[\log\Bigg(1 - \mathbb{D}\Big(\mathbf{x}, \mathbb{G}_{\phi,\theta,\boldsymbol{\mathcal{D}}_0}\big(\mathbf{x}; \mathcal{T}(t; \boldsymbol{\mathcal{D}}_0, \theta)\big)\Big)\Bigg)\Bigg],\end{aligned} \tag{6}$$

where $t_0$ and $t_{-1}$ are hyperparameters defining the range that $t$ is sampled from.

***Conditional on labels.*** As pointed out in [7], cGANs are good at conditional image generation with discrete labels, but can struggle at continuous labels. In our method, without leveraging any customized training schemes as in [7], we show that the continuous image generation with standard empirical risk minimization can be significantly improved by placing an assumption on the graduate appearance change across continuous labels, and trained with continuous atom generation given continuous labels as the input. Formally, we define the scalar label as $x$ sampled continuously from $x \sim p(x)$, and plug in the continuous atom generator to standard cGAN to model the desirable target distribution $p(\mathbf{y}|x)$. The image generation is then achieved as $\mathbb{G}_{\phi,\theta,\boldsymbol{\mathcal{D}}_0}(\mathbf{z}; \mathcal{T}(\alpha x; \boldsymbol{\mathcal{D}}_0, \theta))$, with an input vector $\mathbf{z} \sim \mathcal{N}(0, \mathbf{I})$ carrying randomness as in standard cGAN. Comparing to (6), the input to the continuous atom generator is replaced as $\alpha x$, where $\alpha$ is a predefined scalar that linearly scales the input continuous label. And we use a standard discriminator with each input consist of an image and a label. The training can be realized as

$$\begin{aligned}\min_{\mathbb{G}} \max_{\mathbb{D}} \mathcal{L}(\mathbb{D}, \mathbb{G}) =& \mathbb{E}_{x\sim p(x), \mathbf{y}\sim p(\mathbf{y}|x)}\big[\log\big(\mathbb{D}(x, \mathbf{y})\big)\big] + \\ & \mathbb{E}_{x\sim p(x), \mathbf{z}\sim \mathcal{N}(0,\mathbf{I})}\Bigg[\log\Bigg(1 - \mathbb{D}\Big(x, \mathbb{G}_{\phi,\theta,\boldsymbol{\mathcal{D}}_0}\big(\mathbf{z}; \mathcal{T}(\alpha x; \boldsymbol{\mathcal{D}}_0, \theta)\big)\Big)\Bigg)\Bigg].\end{aligned} \tag{7}$$

***Conditional on both images and labels.*** For image-to-image translation tasks, a few works condition on both images and labels to achieve scalable translation while introducing diversity. Specifically, [38] and [4] have imposed the domain attributes to the input and the generator, respectively, while conditioning on input images. However, the domain attributes they condition on are all categorical labels, and none of them can adopt continuous or raw format (e.g, float number) labels to capture gradual changes w.r.t the input images. Our work imposes the continuous labels to the atom generator to explicitly control the space of convolutional filters, while also conditioning image generation on input images, thus generates images with smooth changes from the source image. To the best of our knowledge, our method is the first work to present gradually changing image-to-image translation results conditioned on both images and continuous labels. Formally, having both $\mathbf{x} \sim p(\mathbf{x})$ and $x \sim p(x)$, which refer to the input image and the scalar label, respectively, we employ our continuous atom generator, $\mathcal{T}(\alpha x; \boldsymbol{\mathcal{D}}_0, \theta)$, to get the desirable target distribution $p(\mathbf{y}|x, \mathbf{x})$. We then formulate our image generator as $\mathbb{G}_{\phi,\theta,\boldsymbol{\mathcal{D}}_0}(\mathbf{x}; \mathbf{z}; \mathcal{T}(\alpha x; \boldsymbol{\mathcal{D}}_0, \theta))$. The training can be realized as

$$\begin{aligned}\min_{\mathbb{G}} \max_{\mathbb{D}} \mathcal{L}(\mathbb{D}, \mathbb{G}) =& \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x}), \mathbf{y}\sim p(\mathbf{y}|\mathbf{x}, x)}\big[\log\big(\mathbb{D}(\mathbf{x}, \mathbf{y})\big)\big] + \\ & \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x}), x\sim p(x)}\Bigg[\log\Bigg(1 - \mathbb{D}\Big(\mathbf{x}, \mathbb{G}_{\phi,\theta,\boldsymbol{\mathcal{D}}_0}\big(\mathbf{x}; \mathcal{T}(\alpha x; \boldsymbol{\mathcal{D}}_0, \theta)\big)\Big)\Bigg)\Bigg].\end{aligned} \tag{8}$$

Figure 4: Continuous image-to-image translation. The network is trained without any auxiliary loss functions or regularizations. From top to bottom, the image to image translation tasks are: nights → days, edges → handbags, and edges → shoes.

Table 1: Quantitative results on image generation conditional on images only.

| Methods | Labels→Facedee | | Map→Satellite | | Night→Day | | Edge→Shoe | | Edge→Handbag | |
|---|---|---|---|---|---|---|---|---|---|---|
| | LPIPS ↑ | FID ↓ | LPIPS ↑ | FID ↓ | LPIPS ↑ | FID ↓ | LPIPS ↑ | FID ↓ | LPIPS ↑ | FID ↓ |
| BicycleGAN | 0.1413 | 98.85 | 0.1150 | 145.78 | 0.103 | 120.63 | 0.139 | 72.49 | 0.184 | 96.28 |
| MSGAN | 0.1894 | 92.84 | 0.2189 | 152.43 | 0.176 | 107.90 | 0.167 | **60.28** | 0.228 | 89.96 |
| BasisGAN | **0.2648** | 88.70 | **0.2417** | **35.54** | **0.184** | **102.56** | **0.242** | 64.17 | 0.350 | 88.76 |
| Ours | 0.2274 | **87.75** | 0.2217 | 55.76 | 0.170 | 106.95 | 0.237 | 61.55 | **0.361** | **85.57** |

# 4 Experiments

In this section, we present results and comparisons in two experimental settings. We start with continuous image-to-image translation tasks where we show that by directly plugging in the proposed continuous atom generator, we can either condition the generated images solely on images, or on both images and labels, where each of the cases refer to (6) and (8), respectively. In both cases, we quantitatively and qualitatively show that our method achieves continuous image generation with high fidelity and diversity. Then, we proceed to image generation tasks conditional on continuous labels following (7). We compare the proposed method with [7] and [23] on the face generation experiments. Implementation details are in Appendix Section B. More qualitative results are in Appendix Section C.

## 4.1 Continuous Image-to-image Translation

The introduced continuous atom generation can be naturally applied to the image-to-image translation experiments [39] as a simple plug-and-play module. Specifically, in image-to-image translation tasks with paired samples, we adopt a ResNet [11] based generator network following the implementation of Pix2Pix [15]. All the convolutional filters of the ResNet blocks are decomposed into atom-coefficients where individual atom of each block is generated continuously with neural ODE. Note that a single neural ODE is utilized throughout the entire network, enforcing the network to be governed by a single continuous dynamic. These settings are also applied in unpaired image-to-image translation tasks, where we replace our baseline model from the Pix2Pix to ResNet based CycleGAN [46].

**Continuous image-to-image translation conditional on images.** We present results on various datasets including edges → handbags, edges → shoes, maps → satellite, night → days, and labels → facades. Comparing to previous methods working on the same datasets, our method does not focus on generating images with strong diversity. Instead, we show that by plugging-in the introduced continuous filter atoms, images with smooth intermediate transitions can be obtained by directly training the network without any additional supervision and regularizations. As shown qualitatively in Figure 4 and quantitatively in Table 1, this smooth transitions in the generated appearance can naturally lead to satisfying diversity without compromising the quality of the generated images and the correspondence to the input conditions.

**Continuous image-to-image translation conditional on regression labels.** We present results of image-to-image translation tasks conditional on regression labels on a variety of datasets and experimental settings. Following (8), the generated images are conditional on explicitly given regression labels, while keeping correspondence to the input image. We start with experiments on image-to-image translation on paired samples, and then proceed to the unpaired samples.

***Paired samples.*** In this experiment, we plug in our continuous filter atoms to the Pix2Pix [15] model and conduct continuous conditional image-to-image translation tasks on five different synthetic datasets, RC-49 [7], RA-20, RCA-20, RT-20, RL-20, where objects in each dataset are rendered to rotate between $0^\circ$ and $360^\circ$ with $0.1^\circ$ interval. Since there are very few datasets of 3-D objects with

Table 2: Quantitative results on image generation conditional on both images and labels under the configuration of image-to-image translation tasks with paired inputs. Diversity and fidelity are measured using LPIPS and FID, respectively. As Pix2Pix is a deterministic model, LPIPS scores for Pix2Pix are all marked as '-', and all the scores resulted in zero.

| Datasets | RC49 | | RCA20 | | RA20 | | RL20 | | RT20 | |
|---|---|---|---|---|---|---|---|---|---|---|
| Methods | Pix2Pix | Ours | Pix2Pix | Ours | Pix2Pix | Ours | Pix2Pix | Ours | Pix2Pix | Ours |
| LPIPS ↑ | - | 0.1127 | - | 0.0931 | - | 0.068 | - | 0.1089 | - | 0.1213 |
| FID ↓ | 251.49 | **195.01** | 218.88 | **85.06** | 165.01 | **73.82** | 227.04 | **124.83** | 206.49 | **70.92** |

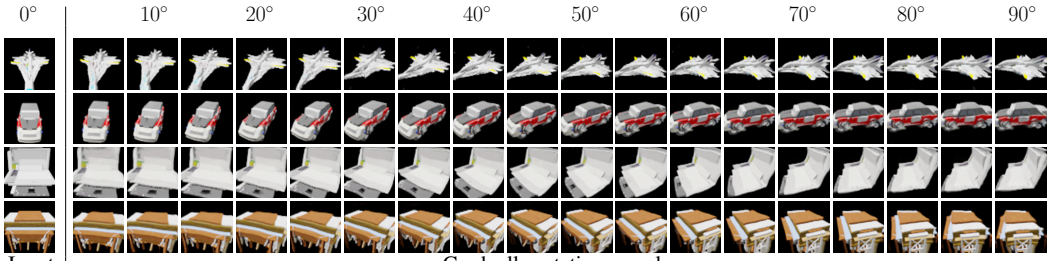


Figure 5: Continuous image-to-image translation conditional on both images and labels with paired samples. From top to bottom, the trained datasets are: RA-20, RCA-20, RL-20, and RT-20. The generated images of all datasets rotate from a degree 0.1 to 89.9 at the interval of 5 degree, along the yaw axis. All models are trained with Pix2Pix by plugging in our continuous atom generator.

continuous labels, we created new datasets RA-20, RCA-20, RT-20, RL-20, by rendering 'Airplanes', 'Cars', 'Tables', 'Laptops' categories of ShapeNet, respectively, following the manuals of RC-49. We add further descriptions of datasets in the Appendix Section B.1. During training, we only use samples with rotation degrees below $90^\circ$, and fix the $0^\circ$ image as the source image with varying target images. We adopt the gap of rotation degree between the source and the target image as an external condition for atom generation. We show in the Figure 5, that the generated images rotate smoothly according to the given rotation degrees. Especially for the results of RC-49, although we have organized the training data in $5^\circ$ interval, it is noteworthy that our method generated interpolation results at the interval of 2.5 degree in the Figure 3. Table 2 also demonstrates that the generated images of our proposed method hold a higher level of fidelity than the baseline.

Table 3: Quantitative results on conditional image generation when image and label were both given as input conditions under the configuration of unpaired image-to-image translation task. We present results on UTKFace [45], CACD2000[2], SteeringAngle, and Cell200. Diversity and fidelity are measured using LPIPS and FID, respectively. The LPIPS scores for the CycleGAN are marked as '-', as the all LPIPS scores resulted in zero, with no meaningful diversity observed.

| Datasets | Cell200 | | SteeringAngle | | UTKFace | | CACD2000 | |
|---|---|---|---|---|---|---|---|---|
| Methods | CycleGAN | Ours | CycleGAN | Ours | CycleGAN | Ours | CycleGAN | Ours |
| LPIPS ↑ | - | 0.1355 | - | 0.0544 | - | 0.0052 | - | 0.0018 |
| FID ↓ | 30.44 | **22.22** | 73.03 | **52.39** | 47.24 | **41.55** | 58.43 | **57.15** |

***Unpaired samples.*** For translation task on unpaired datasets, we present results on UTKFace [45], SteeringAngle, and Cells-200. Note that all the datasets are composed of images with continuous labels, e.g., the age for UTKFace, the angle of the steering wheel of a car for the SteeringAngle, and the number of cells for Cells-200. We add more detailed descriptions on the datasets in the Appendix Section B.1 with additional results of aging face experiments on the CACD2000 [2] at Appendix Section B.2. Since it is intractable to assign separate label to each pair of images under the configuration of unpaired image-to-image translation task, the target domain is divided to a specific interval of labels, so that each sub-group of the target domain can be assigned distinct labels instead. The distinct labels are then directly used as a condition for modeling the continuous subspace of atoms as in (8). Meanwhile, images with certain conditions,.e.g, teenagers for the UTKFace, -65° to -80° for SteeringAngle, one number of cell for Cells-200, are fixed as a source domain input during the training. Figure 6 shows the gradual changing between the condition of target domain. While keeping certain amount of consistency within the input image, smooth transition between generated images are observed. Table 3 also highlights our proposed method's superior fidelity over the baseline model. Additional training details are presented in the Appendix Section B.

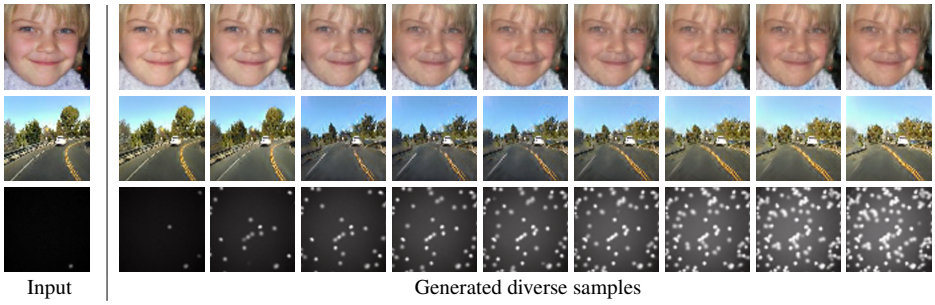


Figure 6: Continuous image-to-image translation conditional on both image and labels with unpaired samples. From top to bottom, the datasets are: UTKFace (young $\rightarrow$ old), SteeringAngle (steering angle at -80° $\rightarrow$ steering angle at 80°), and Cell200 (one cell $\rightarrow$ two-hundred cells). All models are trained with CycleGAN by plugging in our continuous atom generator.

Figure 7: Continuous image generation conditional on regression labels. From left to right, each column are images generated with the same atoms sampled from the continuous atom space and model the appearance of face at a particular age from 0 to 60. By fixing $\mathbf{z}$ in (7), each row shows the graduate appearance changes w.r.t. continuously sampled atoms.

### 4.2 Conditional Image Generation with Regression Labels

We present face image generation with condition only on a regression label specifying the age. The training objective follows (7) strictly. We construct a conditional image generation network by replacing the Adaptive Instance Normalization (AdaIN) [12] layers with standard batch normalization layers, and replace convolutional filters with the proposed continuous atom generation networks. As shown in the qualitative results in Figure 7, by fixing the input latent code $\mathbf{z}$, and changing only the condition scalar $x$, we are able to generate sequences of images with graduate age changes. And switching $\mathbf{z}$ allows changing the identity of the face sequence. Comparable quantitative results in Table 4 with CcGAN [7] further demonstrate the effectiveness.

Table 4: Quantitative results on conditional image generation when the label is given as an input condition. We present results on UTKFace [45] with age as regression labels.

| Dataset | UTKFace | | |
|---|---|---|---|
| Methods | cGAN | CcGAN | Ours |
| Intra-FID ↓ | 4.516 | 0.425 | 0.432 |
| NIQE ↓ | 2.315 | 1.725 | 1.749 |
| Diversity ↑ | 0.254 | 1.298 | 1.321 |
| Label Score ↓ | 11.087 | 7.452 | 7.399 |

## 5 Conclusion

In this paper, we presented both theoretically and empirically that, continuous dynamics of convolutional filters can be effectively modeled in the filter subspace by neural ordinary differential equation, to subsequently achieve conditional image generation with graduate appearance changes. The introduced continuous filter atom generation enables continuous image generation conditional on images, labels, or both. We demonstrated its effectiveness using both superior quantitative and qualitative results.

# References


[1] Dongsheng An, Yang Guo, Min Zhang, Xin Qi, Na Lei, and Xianfang Gu. Ae-ot-gan: Training gans from data specific latent distribution. In *European Conference on Computer Vision*, pages 548–564. Springer, 2020.

[2] Bor-Chun Chen, Chu-Song Chen, and Winston H Hsu. Cross-age reference coding for age-invariant face recognition and retrieval. In *European conference on computer vision*, pages 768–783. Springer, 2014.

[3] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural ordinary differential equations. *arXiv preprint arXiv:1806.07366*, 2018.

[4] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8789–8797, 2018.

[5] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8188–8197, 2020.

[6] Earl A Coddington and Norman Levinson. *Theory of ordinary differential equations*. Tata McGraw-Hill Education, 1955.

[7] Xin Ding, Yongwei Wang, Zuheng Xu, William J Welch, and Z Jane Wang. Ccgan: Continuous conditional generative adversarial networks for image generation. *arXiv preprint arXiv:2011.07466*, 2020.

[8] John R Dormand and Peter J Prince. A family of embedded runge-kutta formulae. *Journal of computational and applied mathematics*, 6(1):19–26, 1980.

[9] Rui Gong, Wen Li, Yuhua Chen, and Luc Van Gool. Dlow: Domain flow for adaptation and generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2477–2486, 2019.

[10] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Advances in Neural Information Processing Systems*, 2014.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[12] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1501–1510, 2017.

[13] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 172–189, 2018.

[14] Zhizhong Huang, Shouzhen Chen, Junping Zhang, and Hongming Shan. Pfa-gan: Progressive face aging with generative adversarial network. *IEEE Transactions on Information Forensics and Security*, 2020.

[15] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[16] David Kanaa, Vikram Voleti, Samira Kahou, Christopher Pal, and Mila CIFAR. Simple video generation using neural odes. In *Annu. Conf. Neural Inf. Process. Syst*, 2019.

[17] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[18] Wilhelm Kutta. Beitrag zur naherungsweisen integration totaler differentialgleichungen. *Z. Math. Phys.*, 46:435–453, 1901.

[19] Hsin-Ying Lee, Hung-Yu Tseng, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Diverse image-to-image translation via disentangled representations. In *Proceedings of the European conference on computer vision (ECCV)*, pages 35–51, 2018.

[20] Wallace Lira, Johannes Merz, Daniel Ritchie, Daniel Cohen-Or, and Hao Zhang. Ganhopper: Multi-hop gan for unsupervised image-to-image translation. In *European Conference on Computer Vision*, pages 363–379. Springer, 2020.

[21] Qi Mao, Hsin-Ying Lee, Hung-Yu Tseng, Siwei Ma, and Ming-Hsuan Yang. Mode seeking generative adversarial networks for diverse image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1429–1437, 2019.

[22] Qi Mao, Hung-Yu Tseng, Hsin-Ying Lee, Jia-Bin Huang, Siwei Ma, and Ming-Hsuan Yang. Continuous and diverse image-to-image translation via signed attribute vectors. *arXiv preprint arXiv:2011.01215*, 2020.

[23] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[24] Takeru Miyato and Masanori Koyama. cgans with projection discriminator. *arXiv preprint arXiv:1802.05637*, 2018.

[25] Aaron van den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. *arXiv preprint arXiv:1601.06759*, 2016.

[26] Yingxue Pang, Jianxin Lin, Tao Qin, and Zhibo Chen. Image-to-image translation: Methods and applications. *arXiv preprint arXiv:2101.08629*, 2021.

[27] Sunghyun Park, Kangyeol Kim, Junsoo Lee, Jaegul Choo, Joonseok Lee, Sookyung Kim, and Edward Choi. Vid-ode: Continuous-time video generation with neural ordinary differential equation. *arXiv preprint arXiv:2010.08188*, 2020.

[28] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2536–2544, 2016.

[29] Pengda Qin, Weiran Xu, and William Yang Wang. Dsgan: Generative adversarial training for distant supervision relation extraction. *arXiv preprint arXiv:1805.09929*, 2018.

[30] Qiang Qiu, Xiuyuan Cheng, Guillermo Sapiro, et al. DCFNet: Deep neural network with decomposed convolutional filters. In *International Conference on Machine Learning*, pages 4198–4207. PMLR, 2018.

[31] Andrés Romero, Pablo Arbeláez, Luc Van Gool, and Radu Timofte. Smit: Stochastic multi-label image-to-image translation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[32] Yulia Rubanova, Ricky TQ Chen, and David Duvenaud. Latent odes for irregularly-sampled time series. *arXiv preprint arXiv:1907.03907*, 2019.

[33] Patsorn Sangkloy, Jingwan Lu, Chen Fang, Fisher Yu, and James Hays. Scribbler: Controlling deep image synthesis with sketch and color. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5400–5409, 2017.

[34] Paul Smolensky. Information processing in dynamical systems: Foundations of harmony theory. Technical report, Colorado Univ at Boulder Dept of Computer Science, 1986.

[35] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. In *Advances in Neural Information Processing Systems*, pages 3483–3491, 2015.

[36] Aaron Van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, et al. Conditional image generation with pixelcnn decoders. In *Advances in Neural Information Processing Systems*, pages 4790–4798, 2016.

[37] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[38] Yaxing Wang, Abel Gonzalez-Garcia, Joost van de Weijer, and Luis Herranz. Sdit: Scalable and diverse cross-domain image translation. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 1267–1276, 2019.

[39] Ze Wang, Xiuyuan Cheng, Guillermo Sapiro, and Qiang Qiu. Stochastic conditional generative networks with basis decomposition. *International Conference of Learning Representation*, 2020.

[40] Po-Wei Wu, Yu-Jing Lin, Che-Han Chang, Edward Y Chang, and Shih-Wei Liao. Relgan: Multi-domain image-to-image translation via relative attributes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5914–5922, 2019.

[41] Wenqi Xian, Patsorn Sangkloy, Varun Agrawal, Amit Raj, Jingwan Lu, Chen Fang, Fisher Yu, and James Hays. Texturegan: Controlling deep image synthesis with texture patches. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8456–8465, 2018.

[42] Taihong Xiao, Jiapeng Hong, and Jinwen Ma. Dna-gan: Learning disentangled representations from multi-attribute images. *arXiv preprint arXiv:1711.05415*, 2017.

[43] Xiaoming Yu, Yuanqi Chen, Thomas Li, Shan Liu, and Ge Li. Multi-mapping image-to-image translation via learning disentanglement. *arXiv preprint arXiv:1909.07877*, 2019.

[44] Jianfu Zhang, Yuanyuan Huang, Yaoyi Li, Weijie Zhao, and Liqing Zhang. Multi-attribute transfer via disentangled representation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9195–9202, 2019.

[45] Zhifei Zhang, Yang Song, and Hairong Qi. Age progression/regression by conditional adversarial autoencoder. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5810–5818, 2017.

[46] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.

# Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [No]
   (c) Did you discuss any potential negative societal impacts of your work? [No]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [Yes] The proof of Theorem 1 is provided in the Appendix.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [No]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]