Dual Projection Generative Adversarial Networks for Conditional Image Generation

Ligong Han,[1] Martin Renqiang Min,[2] Anastasis Stathopoulos,[1]
Yu Tian,[1] Ruijiang Gao,[3] Asim Kadav,[2] Dimitris Metaxas[1]
[1]Department of Computer Science, Rutgers University [2]NEC Labs America
[3]McCombs School of Business, The University of Texas at Austin
lh599@rutgers.edu, renqiang@nec-labs.com, as2947@cs.rutgers.edu
yt219@cs.rutgers.edu, ruijiang@utexas.edu, asim@nec-labs.com, dnm@cs.rutgers.edu

Abstract

*Conditional Generative Adversarial Networks (cGANs) extend the standard unconditional GAN framework to learning joint data-label distributions from samples, and have been established as powerful generative models capable of generating high-fidelity imagery. A challenge of training such a model lies in properly infusing class information into its generator and discriminator. For the discriminator, class conditioning can be achieved by either (1) directly incorporating labels as input or (2) involving labels in an auxiliary classification loss. In this paper, we show that the former directly aligns the class-conditioned fake-and-real data distributions $P(image|class)$ (*data matching*), while the latter aligns data-conditioned class distributions $P(class|image)$ (*label matching*). Although class separability does not directly translate to sample quality and becomes a burden if classification itself is intrinsically difficult, the discriminator cannot provide useful guidance for the generator if features of distinct classes are mapped to the same point and thus become inseparable. Motivated by this intuition, we propose a Dual Projection GAN (P2GAN) model that learns to balance between* data matching *and* label matching. *We then propose an improved cGAN model with Auxiliary Classification that directly aligns the fake and real conditionals $P(class|image)$ by minimizing their f-divergence. Experiments on a synthetic Mixture of Gaussian (MoG) dataset and a variety of real-world datasets including CIFAR100, ImageNet, and VGGFace2 demonstrate the efficacy of our proposed models.*

1. Introduction

Generative Adversarial Networks (GANs) [7] are an algorithmic framework that allows implicit generative modeling of data distribution from samples [29]. It has attracted great attention due to its ability to model very high-dimensional data, such as images or videos and to produce sharp and faithful samples [16, 38, 3]. Conditional GAN (cGAN) [25, 33, 28] is an extension of GAN that utilizes the label information and aims to learn the joint distribution of data and label. Thanks to its ability to control over the generative process by conditioning on labels, it has been widely adopted in real-world problems including class-conditional image generation [31, 33, 5], text-to-image generation [35, 39], image-to-image generation [14, 40], text-to-video synthesis [22, 1, 10, 9], domain adaptation [13, 26] *etc*.

Different conditional GANs differ in the way how data and labels are incorporated in the discriminator. Class conditioning can be achieved by either (1) conditioning the discriminator directly on labels or their embeddings [25, 14, 28], or by (2) incorporating an auxiliary classification loss in the objective, as in (T)AC-GANs[25, 6]. Recently, cGAN has received a major update that changed the label from being concatenated [25, 35, 39] to being projected [28]. The projection discriminator takes the inner product between the label embedding and the data/image embedding, and remains to be the choice of many state-of-the-art methods [38, 3].

In this paper, we first give insights on projection discriminators. We point out that the success of projection can be explained by its flexible form. By tying real and fake class embeddings as their difference, a projection discriminator can ideally realize two extremes in a single form: (1) matching conditional label distributions $P(\text{class}|\text{image})$ (*label matching*), and (2) matching conditional data distributions $P(\text{image}|\text{class})$ (*data matching*). Moreover, the visualization of data embeddings of trained projection discriminators does not show obvious patterns of class clustering. This suggests that projection may bias towards conditional data matching. Although label matching does not

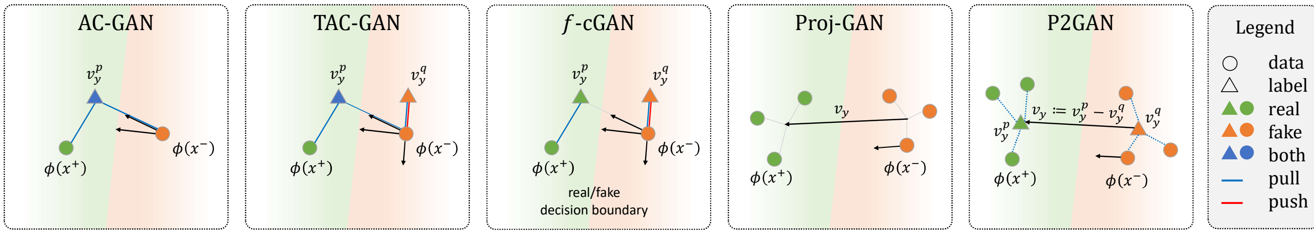


Figure 1: Illustrative figures visualize the learning schema of conditional discriminator losses. The color indicates how real/fake embeddings interact during *discriminator training*. The green/red boundary indicates (unconditional) real/fake decision boundary. Triangles represent class embeddings, and circles indicate image embeddings (*e.g.* v_y^p is blue triangle in AC-GAN since it is class embedding and is trained on both real and fake data). Solid blue and red lines represent pull/push forces respectively. Black arrows indicate the forces that a fake image embedding receives.

directly translate to the fidelity of generated samples, it is still desirable for generating high-quality images. For example, the discriminator would not provide useful guidance for the generator if features of distinct classes are mapped to the same point and thus become inseparable.

To this end, we propose a new conditional generative adversarial network, namely *Dual Projection GAN* (P2GAN). The main feature of our design is to inherit the flexibility of projection while performing explicit label matching. Realized by auxiliary classification losses, label matching enables the discriminator to exploit useful information for the generator. However, if such task is intrinsically difficult (such as ImageNet [36]), label matching becomes a burden. In an extreme case when the auxiliary classification task failed completely, (T)AC-GANs will degrade to an unconditional GAN while P2GAN degrades to a projection GAN. We also present adaptive approaches to weighing data matching and label matching. Furthermore, we respectively propose two variants for explicit data matching and label matching: (1) direct data matching GAN (DM-GAN), (2) and f-cGAN which aligns the fake and real conditionals $P(\text{class}|\text{image})$ by minimizing their f-divergence. Finally, we conduct extensive experiments on various datasets to show the efficacy of the proposed models.

2. Related Work

Projection. The key idea of Proj-GAN is to tie parameters of AC and twin AC. This design brings flexibility in modeling data matching and label matching. We leverage its flexible design in our P2GAN model. The same projection form is considered in projection-based MINE [2, 11].

Multi-task learning. Training GANs with both discrimination loss and classification loss can be viewed as a multi-task learning problem. [18] propose a principled approach [18] that weighs multiple loss functions by considering the homoscedastic uncertainty of each task. The proposed P2GAN-w model is built upon this notion.

f-divergence in GANs. [32] formulated a family of GANs into a variational f-divergence minimization framework [32]. Their proposed f-GAN is for marginal matching (unconditional GAN) while our f-cGAN aims to minimize the f-divergence between real and fake conditionals and does not optimize its dual form.

Twin auxiliary classifiers. TAC-GAN corrects the bias in AC-GANs by introducing a twin AC and maximizing its classification loss on generated samples. A binary version of twin AC has been introduced as an Anti-Labeler in CausalGAN [19]. CausalGAN has observed unstable training behavior after introducing such binary twin ACs. Rob-GAN [23] splits the categorical classification losses of real and fake data in AC-GAN, however, it does not have a twin AC (dual projections) and does not aim to balance label-matching and data-matching.

3. Method

3.1. Background

Throughout the paper, we denote data-label pairs as $\{x_i, y_i\}_{i=1}^n \subseteq \mathcal{X} \times \mathcal{Y}$ drawn from the joint distribution P_{XY}, where x is an image (or other forms of data) and y is its label, *i.e.*, $\mathcal{Y} = \{1, \ldots, K\}$. A generator is trained to transform samples $z \sim P_Z$ from a canonical distribution conditioned on labels to match the real data distributions. *Real* distributions are denoted as P and *generated* distributions are denoted as Q. We also denote real and fake data as $x^+ \sim P_X$ and $x^- \sim Q_X$, respectively. The discriminator of a conditional GAN learns to distinguish samples drawn from the joint distribution P_{XY} and Q_{XY}. Its objectives can be written as:

$$
\begin{aligned}
L_D =& \mathbb{E}_{x,y\sim P_{XY}} \mathcal{A}(-\tilde{D}(x,y)) + \qquad (1)\\
& \mathbb{E}_{z\sim P_Z, y\sim Q_Y} \mathcal{A}(\tilde{D}(G(z,y),y)), \quad \text{and}\\
L_G =& \mathbb{E}_{z\sim P_Z, y\sim Q_Y} \mathcal{A}(-\tilde{D}(G(z,y),y)). \qquad (2)
\end{aligned}
$$

Here $\mathcal{A}$ is the *activation function* and $\tilde{D}$ is the *logit* or discriminator's output before activation. Note that choosing

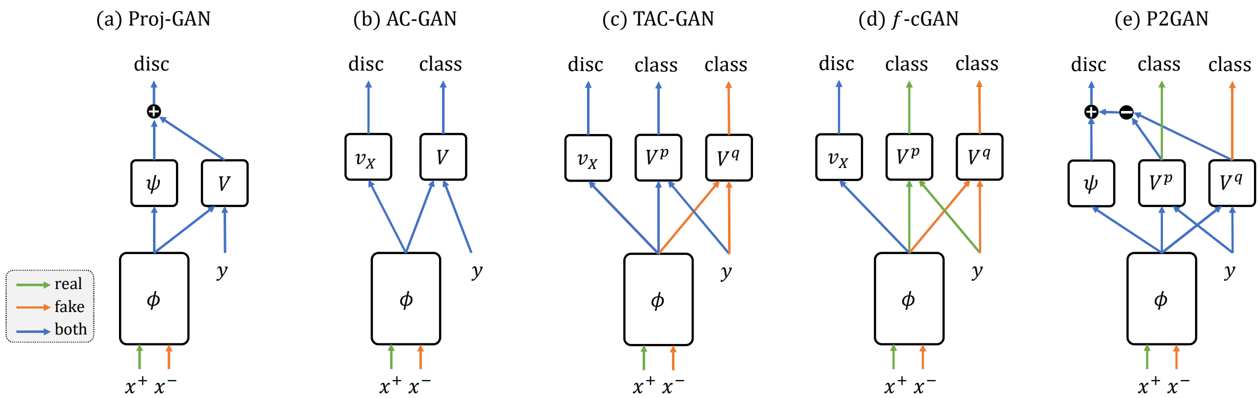


Figure 2: Discriminator models for conditional GANs. Solid green lines represent the flow path of real data, red line represent the flow path of generated data and blue lines mean both real and generated data flow through them. x^+ represents real images, x^- are generated images and y are labels. Matrix V^p and V^q are the collection of class embeddings $\{v_y^p\}$ and $\{v_y^q\}$ for twin auxiliary classifiers, respectively.

$\mathcal{A}(t) = softplus(t) = \log(1 + \exp(t))$ recovers the original GAN formulation[7]. With this activation function, the logit of an optimal discriminator in Equation 1 can be decomposed in two ways,

$$\tilde{D}^*(x, y) = \underbrace{\log \frac{P(x)}{Q(x)}}_{\text{Marginal Matching}} + \underbrace{\log \frac{P(y|x)}{Q(y|x)}}_{\text{Label Matching}} \tag{3}$$

$$= \underbrace{\log \frac{P(x|y)}{Q(x|y)}}_{\text{Data Matching}} + \log \frac{P(y)}{Q(y)}. \tag{4}$$

From Equation 3, one can derive the logit of a *projection* discriminator [28],

$$\tilde{D}(x, y) = v_y^{\mathrm{T}} \phi(x) + \psi(\phi(x)), \tag{5}$$

where $\phi(\cdot)$ is the image embedding function, v_y is embedding of class y, and ψ collects the residual terms. $v_y := v_y^p - v_y^q$ is the difference of real and fake class embeddings.

3.2. Projection and Dual Projection GANs

Effectiveness of tying class embeddings. The key idea of a projection discriminator [28] is to tie the parameters v_y^p and v_y^q into a single v_y. Intuitively, tying embeddings allows a projection discriminator to turn the problem of learning categorical decision boundaries to learning a relative translation vector for each class. The latter is much easier than the former. Intuitively, it is in general easier to learn a relative comparison than to learn each parameter in an absolute scale. For example, learning v_y^p is a hard problem by itself when training on large scale datasets like ImageNet [36].

Without loss of generality, let's assume $\psi(\cdot)$ as a linear function v_ψ. From Equation 1 we can see that L_D is trying to maximize $(v_y + v_\psi)^{\mathrm{T}} \phi(x^+)$ and to minimize $(v_y + v_\psi)^{\mathrm{T}} \phi(x^-)$. If we approximate *softplus* function by $ReLU = \max(0, \cdot)$, we observe that the loss is large when x^+ and x^- are mis-classified, under which $L_D \approx (v_y + v_\psi)^{\mathrm{T}}(\phi(x^-) - \phi(x^+))$. Intuitively, the learning procedure of Proj-GAN can be understood as performing two alternative steps,

$$\text{D-step}: \ \text{Align } (v_y + v_\psi) \text{ with } (\phi(x^+) - \phi(x^-));$$
$$\text{G-step}: \ \text{Move } \phi(x^-) \text{ along } (v_y + v_\psi).$$

This shows that by tying parameters, Proj-GAN is able to directly perform *data matching* (align $Q(x|y)$ with $P(x|y)$) without explicitly enforcing *label matching*.

Enforcing label matching. Ideally, v_y should recover the difference between underlying v_y^p and v_y^q, but there is no explicit constraint to enforce such property. To do this, we start by untying the class embeddings $v_y := v_y^p - v_y^q$. Then, we encourage V^p and V^q to learn conditional distributions $p(y|x)$ and $q(y|x)$, respectively*. This is usually done by minimizing cross-entropy losses with a $softmax$ function,

$$L_{mi}^p = -v_y^{p\mathrm{T}} \phi(x^+) + \log \sum_{y'} \exp(v_{y'}^{p\mathrm{T}} \phi(x^+)), \text{ and}$$
$$L_{mi}^q = -v_y^{q\mathrm{T}} \phi(x^-) + \log \sum_{y'} \exp(v_{y'}^{q\mathrm{T}} \phi(x^-)). \tag{6}$$

Note that the two classifiers V^p and V^q are trained on real and fake data respectively. This is different from AC-GAN and TAC-GAN where the (first) discriminator is trained on both x^+ and x^-. This is shown in Figure 1 and 2.

Dual Projection GAN (P2GAN). The discriminator and

*Matrix V is the collection of class embeddings v_y for all categories.

generator losses for P2GAN is given as follows,

$$L_D^{P2} = L_D(\tilde{D}) + L_{mi}^p + L_{mi}^q, \tag{7}$$
$$\text{and } L_G^{P2} = L_G(\tilde{D}),$$
$$\text{with } \tilde{D} = (v_y^p - v_y^q)^{\mathrm{T}}\phi(x) + \psi(\phi(x))$$

The proposed method has untied projection vectors, thus we term the model Dual Projection GAN, or P2GAN. Note that both $L_D(\tilde{D})$ and L_{mi}^p contain parameter V^p, and similarly, both $L_D(\tilde{D})$ and L_{mi}^q contain parameter V^q. It is also worth mentioning that similar to the Proj-GAN, P2GAN only proposes a form of the logit while leaves the activation function free of choices.

We observe that when cross-entropy losses are removed, P2GAN is equivalent to Proj-GAN except that v_y is over-parameterized as $v_y^p - v_y^q$. This suggests that when class separability is intrinsically difficult to model and the two cross-entropy losses are not effectively optimized, P2GAN reduces to a Proj-GAN. In this case, a model that relies on *marginal-conditional* [21] decomposition, such as an AC-GAN or a TAC-GAN, degrades to an unconditional GAN instead. This illustrates that the proposed P2GAN *implicitly* balances data matching and label matching by inheriting the Proj-GAN. We can also *explicitly* weigh L_{mi}^p and L_{mi}^q.

Weighted Dual Projection GAN (P2GAN-w). From a multi-task learning perspective [18], it should be beneficial to let the model learn to weigh *data matching* and *label matching*. Here we consider adding a "gate" between the two losses,

$$L_D^{P2w} = L_D + \lambda \cdot (L_{mi}^p + L_{mi}^q). \tag{8}$$

We tested three variants of weighted P2GAN: (1) exponential decay, (2) scalar valued, and (3) amortised.

P2GAN-d. The simplest way is to define λ as a decaying factor. We set $\lambda = e^{-t/T}$, t is the number of iterations during training time.

P2GAN-s. Let $\lambda \geq 0$ be a learnable parameter and initialized as 1. When $\lambda = 0$, P2GAN-s reduces to a Proj-GAN. When $\lambda > 0$, class separation is explicitly enforced. The data matching task is intuitively much easier than the label matching task in the early stage of training and λ would quickly vanish. To accommodate this, we follow [18] and add a penalty term on λ (P2GAN-sp),

$$L_D^{P2sp} = L_D + \lambda \cdot (L_{mi}^p + L_{mi}^q) - \frac{1}{2}\log\lambda. \tag{9}$$

P2GAN-a. We also experimented with learning amortized homoscedastic weights for each data point. $\lambda(x) \geq 0$ is then a function of x (P2GAN-a). Since $\lambda(x)$ are per-sample weights, in Equation 8 it should be inside of the expectation. We defer its lengthy definition in Supplementary. Likewise, a penalty term can be added (P2GAN-ap). Also, when loss terms involve non-linearity in the mini-batch expectation, for example, the log in MINE [2], any type of "linearization" tricks can be applied [30].

Comparisons of different weighting strategies are given in experiments Table 1. In practice we find $\lambda(x)$ with penalty works best. In following Experiments, P2GAN-w stands for P2GAN-ap, if not specified.

3.3. Theoretical Analysis

As discussed previously, a projection discriminator defined in Equation 5 can flexibly transit between two decomposition forms. Here we discuss two variants of conditional GAN models that perform *only* data matching *or* label matching.

cGAN that performs data matching (DM-GAN). Setting $\psi(\cdot)$ as zero[†], Proj-GAN can be viewed as a weighted sum of K unconditional GAN objectives with binary cross-entropy losses (where K is the number of classes):

Proposition 1. *When $\psi = 0$, a Proj-GAN reduces to K unconditional GANs, each of them minimizes the Jensen-Shannon divergence between $P_{X|y}$ and $Q_{X|y}$ with mixing ratio $\{\frac{P(y)}{P(y)+Q(y)}, \frac{Q(y)}{P(y)+Q(y)}\}$. Its value function can be written as,*

$$\mathbb{E}_{P_Y}\left\{\mathbb{E}_{P_{X|Y}} \log D(x|y) + \frac{Q_y}{P_y}\mathbb{E}_{Q_{X|Y}} \log(1 - D(x|y))\right\}.$$

We observe a slight improvement in terms of FID score from our early experiment on CIFAR100[‡], which might suggest that Proj-GAN biases towards data matching than label matching. DM-GAN is also the cGAN implementation in [24, 17].

cGAN that performs label matching (f-cGAN). At the other end of the spectrum, if we explicitly model the marginal matching and label matching terms in Equation 3, we arrive at a (T)AC-GAN-like model which follows the *marginal-conditional* decomposition. To be precise, marginal matching can be achieved via (unconditional) GAN loss, and label matching can be enforced by minimizing their divergence.

Proposition 2. *Given a generator G, if cross entropy losses L_{mi}^p and L_{mi}^q are minimized optimally, then the difference of two losses evaluated at fake data equals the reverse KL-divergence between $P_{Y|X}$ and $Q_{Y|X}$,*

$$L_{mi}^p(x^-) - L_{mi}^q(x^-) = \mathbb{E}_{Q_X} KL(Q_{Y|X} \| P_{Y|X}). \tag{10}$$

[†]In practice spectral normalization [27] is often applied and this null solution for ψ cannot be reached.

[‡]this conclusion is not reflected in Table 7 due to different hyper-parameters and configurations.

From the above proposition we derive a new conditional GAN,

$$\begin{aligned} & L_D^f = L_D(\tilde{D}) + L_{mi}^p(x^+) + L_{mi}^q(x^-), \\ \text{and } & L_G^f = L_G(\tilde{D}) + L_{mi}^p(x^-) - L_{mi}^p(x^-), \\ \text{with } & \tilde{D} = v_X^{\mathrm{T}}\phi(x) + b \end{aligned} \tag{11}$$

In fact, $L_{mi}^p(x^-) - L_{mi}^p(x^-) = \mathbb{E}_{Q_{XY}} - \log \frac{Q^p(y|x)}{Q^q(y|x)}$. By replacing the $-\log$ with function f, we can also generalize the reverse-KL to f-divergence [32], $\mathbb{E}_{Q_{XY}} f\left(\frac{P(y|x)}{Q(y|x)}\right) = \mathbb{E}_{Q_X} D_f(P_{Y|X} \| Q_{Y|X})$. The generator loss then becomes

$$L_G^f = L_G + \mathbb{E}_{Q_{XY}} f(\exp(T^p(x,y) - T^q(x,y))). \tag{12}$$

Here, $T^p = v_y^{p\mathrm{T}}\phi(x) - \log \sum_{y'} \exp(v_{y'}^{p\mathrm{T}}\phi(x))$ is an estimator of $\log P(y|x)$, and similarly T^q is an estimator of $\log Q(y|x)$. Equation 11 and 12 defines a general conditional GAN framework, which we call f-cGAN.

To show that f-cGAN is theoretically sound and Nash equilibrium can be achieved, we provide the following theorem.

Theorem 1. *Denoting P_{XY} and Q_{XY} as the data distribution and the distribution induced by G, their Jensen-Shannon divergence is upper bounded by the following,*

$$\begin{aligned} & JSD(P_{XY}, Q_{XY}) \leq \\ & 2c_1\sqrt{2JSD(P_X, Q_X)} + c_2\sqrt{2KL(P_{Y|X} \| Q_{Y|X}^p)} + \\ & c_2\sqrt{2KL(Q_{Y|X} \| Q_{Y|X}^q)} + c_2\sqrt{2KL(Q_{Y|X}^q \| Q_{Y|X}^p)}. \end{aligned} \tag{13}$$

in the above, constants c_1 and c_2 are upper bounds of $\frac{1}{2}\int |P_{Y|X}(y|x)|\mu(x,y)$ and $\int |Q_X(x)|\mu(x)$, respectively, where μ is a σ-finite measure.

3.4. Comparison with Other Methods

First, dual projection GAN extends a projection GAN by untying class embeddings and enforcing class separability in each domain. Comparing P2GAN with f-cGAN, both of them minimizes $L_{mi}^p(x^+) + L_{mi}^q(x^-)$ in the discriminator loss. As for the generator loss, f-cGAN minimizes $L_{mi}^p(x^-) - L_{mi}^q(x^-)$, while P2GAN enforces label matching via $L_G((v_y^p - v_y^q)^{\mathrm{T}}\phi(x^-) + \psi(\phi(x^-)))$ thus the $LogSumExp$ function is not involved.

By adding a term $L_{mi}^p(x^-)$ to L_D^f in Equation 11, one can recover the TAC-GAN[§]. It is worth noting that this term is required since TAC-GAN aims to chain $\mathbb{E}_{P_X}\mathrm{KL}(P_{Y|X} \| Q_{Y|X}^p)$ and $\mathbb{E}_{Q_X}\mathrm{KL}(Q_{Y|X} \| Q_{Y|X}^p)$ together, while our f-cGAN tries directly align $Q_{Y|X}$ with $P_{Y|X}$. In experiments we show that this simple trick can largely boost performance. Further, by removing all terms related to L_{mi}^q we derive AC-GAN.

Both P2GAN and f-cGAN only model *data-to-class* relations, and *data-to-data* relation is indirectly modeled via class embeddings. While ContraGAN [15] models these two relations directly. It is orthogonal to our method and we leave it for future work.

[§]Note that this term is already omitted in its actual implementation, however, it is indispensable for the theoretical analysis to hold.

Table 1: FID scores of different weighting strategies for P2GAN-w. All models are trained for 80000 iterations on ImageNet, 62000 iterations on CIFAR100, and 50000 iterations on VGGFace200.

Datasets	ImageNet	CIFAR100	VGGFace200
P2GAN	19.22	10.55	23.15
P2GAN-d (T=200)	-	10.51	21.24
P2GAN-d (T=2000)	-	10.35	24.49
P2GAN-s	18.62	**9.03**	20.59
P2GAN-sp	20.68	9.51	20.18
P2GAN-a	-	10.13	20.26
P2GAN-ap	**18.31**	9.82	**18.99**

Table 2: Average rank of Proj-GAN, TAC-GAN, f-cGAN and P2GAN on MoG dataset. For each method, the average ranks of using BCE loss, hinge loss, and all experiments are reported.

Models	Proj-GAN	TAC-GAN*	f-cGAN (ours)	P2GAN (ours)
BCE	3.90	2.45	2.15	**1.50**
Hinge	**1.80**	3.35	2.45	2.40
Overall	2.85	2.90	2.30	**1.95**

Table 3: Max-FID scores for 1D MOG synthetic dataset.

BCE / Hinge	$d_m = 1$	$d_m = 2$	$d_m = 3$	$d_m = 4$	$d_m = 5$
Proj-GAN	0.0059	0.0213	0.0291	0.0339	0.0796
TAC-GAN*	**0.0029**	0.0090	0.0134	**0.0169**	0.0243
f-cGAN	0.0032	**0.0072**	**0.0129**	0.0184	**0.0213**
P2GAN	**0.0026**	**0.0060**	**0.0085**	**0.0124**	**0.0212**
Proj-GAN	**0.0200**	**0.0481**	**0.0733**	0.1253	0.2066
TAC-GAN*	0.0270	0.0887	0.1307	0.1483	0.2285
f-cGAN	**0.0236**	0.0829	0.1187	**0.0847**	**0.1100**
P2GAN	0.0245	**0.0473**	**0.0901**	**0.1076**	**0.1698**

4. Experiments

Datasets. In this section, we first show experimental results for model analysis on an imbalanced subset of CIFAR100 [20] (denoted as CIFAR100IB). Then we then report results of different baselines across various datasets. Finally, ablation studies are presented. We evaluate the distribution matching ability of different models on a synthetic

Table 4: Inception Scores (IS), Fréchet Inception Distances (FID) and the maximum intra FID (max-FID). Our proposed adaptive P2GAN achieves the highest IS, lowest FID and lowest max-FID in most cases. The top-two best performing methods are marked in boldface.

	Proj-GAN			TAC-GAN*			f-cGAN (ours)			P2GAN (ours)			P2GAN-w (ours)		
	IS $\uparrow$	FID $\downarrow$	max-FID $\downarrow$	IS $\uparrow$	FID $\downarrow$	max-FID $\downarrow$	IS $\uparrow$	FID $\downarrow$	max-FID $\downarrow$	IS $\uparrow$	FID $\downarrow$	max-FID $\downarrow$	IS $\uparrow$	FID $\downarrow$	max-FID $\downarrow$
CIFAR100	**9.17 $\pm$ 0.14**	9.98	**186.46**	**8.86 $\pm$ 0.12**	**9.63**	223.24	8.41 $\pm$ 0.09	10.76	215.86	8.55 $\pm$ 0.12	10.56	208.47	8.50 $\pm$ 0.10	**9.84**	**204.69**
ImageNet	16.14 $\pm$ 0.34	22.26	147.69	14.85 $\pm$ 0.27	22.62	257.45	16.44 $\pm$ 0.26	**19.28**	**145.44**	**17.78 $\pm$ 0.41**	19.80	170.40	**17.40 $\pm$ 0.33**	**18.87**	**136.91**
VGGFace200	50.93 $\pm$ 0.86	61.43	239.61	40.78 $\pm$ 0.57	96.06	478.10	109.94 $\pm$ 1.15	29.54	215.50	**148.48 $\pm$ 2.87**	**20.70**	**209.86**	**171.31 $\pm$ 3.44**	**15.70**	**127.43**
VGGFace500	126.19 $\pm$ 1.95	23.57	162.27	150.05 $\pm$ 2.31	19.30	233.00	175.59 $\pm$ 2.46	16.74	**136.21**	**210.75 $\pm$ 1.87**	**12.09**	**130.58**	**182.91 $\pm$ 1.24**	**12.73**	151.66
Average Rank	3.5	4	3	4	3.75	5	3.5	3.25	2.75	**1.75**	2.5	2.5	2.25	**1.5**	**1.75**

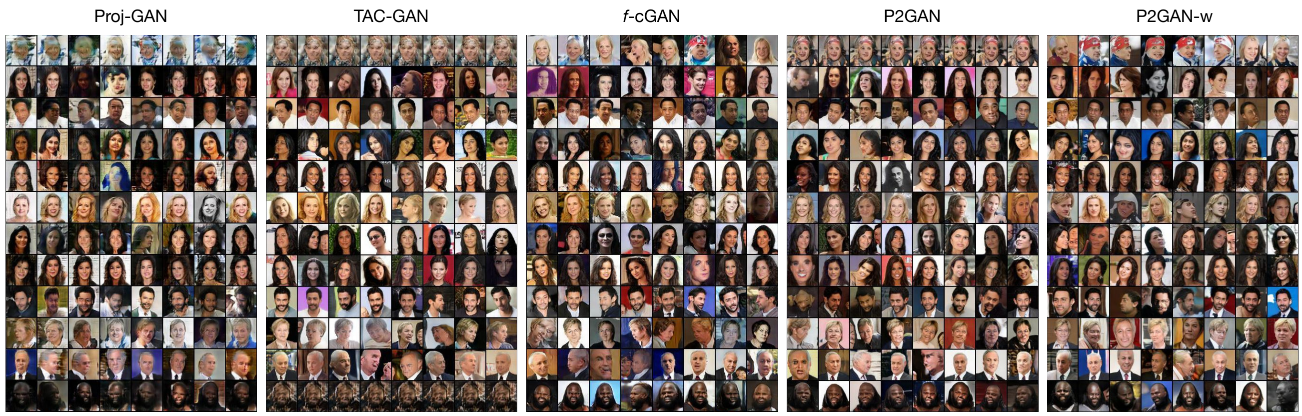


Figure 3: Samples of VGGFace200. Proj-GAN produces blurry faces. TAC-GAN and P2GAN occasionally show mode collapse (first and last row). The adaptive P2GAN model generates diverse and sharp faces.

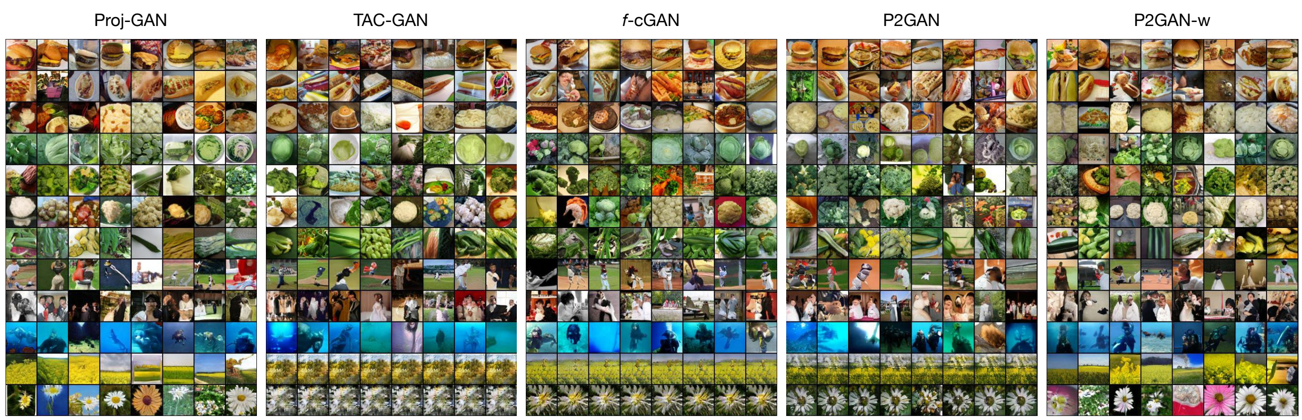
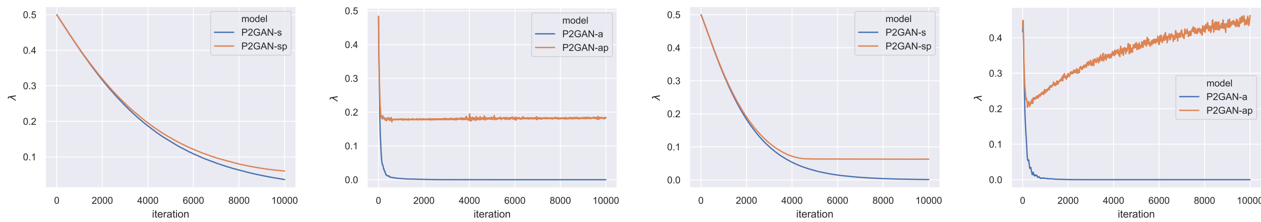


Figure 4: Samples of different methods trained on ImageNet. TAC-GAN, f-cGAN and P2GAN all show mode collapse (last two rows), while Proj-GAN and P2GAN-w generate diverse samples.

Mixture of Gaussian (MoG) dataset [6] and evaluate the image generation performance on CIFAR100, ImageNet [36] and VGGFace2 [4]. Following the protocol in [6], we construct 2 subsets VGGFace200 and VGGFace500.

Baselines. We compare our proposed models with competitive baselines including Proj-GAN and TAC-GAN. We use BigGAN [3] as backbone for baselines. For a fair comparison, we implemented TAC-GAN following the paper [6], and we denote as TAC-GAN* in following experiments. The code is written in PyTorch [34].

Evaluation. Inception Scores (IS) [37] and Fréchet Inception Distances (FID) [12] are reported for quantitative evaluation. We compute intra FIDs [28] between the generated images and dataset images within each class. [28] report the mean value of intra FIDs while here we report the maximum among all classes to better capture mode collapse of trained generative models. Experimental setup and additional results are detailed in Supplementary.



(a) λ on VGGFace200 (b) $\lambda(x)$ on VGGFace200 (c) λ on CIFAR100IB (d) $\lambda(x)$ on CIFAR100IB

Figure 5: Comparison of scalar λ and amortized $\lambda(x)$. Plots show the value of λ as training proceed. Curves are smoothed for better visualization.

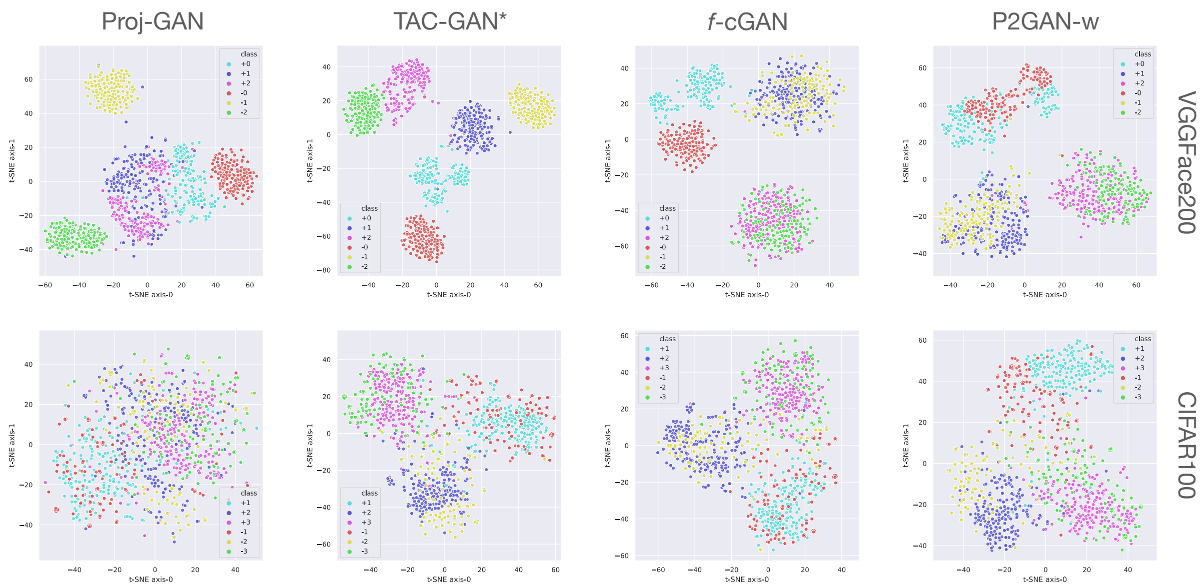


Figure 6: Data embedding visualized by t-SNE. Plot shows 2-D image embedding visualization for baselines at iteration 50000. Positive labels are real image embeddings and negative labels are fake image embeddings. A clear separation between real/fake indicates mode collapse.

Table 5: IS and FID scores on ImageNet at 128 resulution and CIFAR10. We mark '*' to FID and IS values reported in StudioGAN [15].

	ImageNet 128×128		CIFAR10	
	IS $\uparrow$	FID $\downarrow$	IS $\uparrow$	FID $\downarrow$
Proj-GAN	30.73	23.07	*9.85	*8.03
P2GAN	**59.24**	**16.86**	9.76	8.00
P2GAN-w	42.69	19.19	**9.87**	**7.99**

Table 6: Min-LPIPS on VGGFace2 dataset.

	VGGFace200	VGGFace500
Proj-GAN	0.198	0.373
TAC-GAN*	0.009	0.162
f-cGAN	0.233	0.385
P2GAN	0.172	0.376
P2GAN-w	**0.387**	**0.416**

4.1. Model Analysis

Embedding visualization. We visualize the learned image embeddings of different methods, shown in Figure 6. For VGGFace200 dataset, we can see all baselines except for P2GAN-w show clear mode collapse (real and fake image embeddings are well separated) at iteration 50000. We can also observe that Proj-GAN does not enforce class separability. Moreover, comparing these two datasets, VGGFace is more clustered (in classes) than CIFAR100. This verifies our intuition that when the classification task is relatively easy, enforcing label matching helps the discriminator to be trained more powerful and thus provide more informative guidance to the generator.

Learning adaptive weights. As shown in Figure 5, without penalty, λ and $\lambda(x)$ quickly converge to 0 during training. Figure 5-(d) shows an interesting phenomenon (from experiments on CIFAR100IB): The model quickly puts high weight on the discrimination loss L_D at the early stage and weighs it less as training proceeds.

f**-divergence.** Here we consider several commonly used loss functions for f-cGAN and give the training results in Supplementary. We observe that the reverse-KL is the most stable and achieves highest IS and lowest FID. Thus, if not specified, reverse-KL is used in following experiments.

4.2. Comparison of Baselines

Mixture of Gaussian (MoG). We follow protocols in [6] to set up experiments. The real data consists of three 1-D Gaussian with standard deviations $\sigma_0 = 1$, $\sigma_1 = 2$, and $\sigma_2 = 3$, and their means are equally spaced with interval d_m ranging from 1 to 5. Average ranks of Maximum Mean Discrepancy (MMD) [8] are reported in Table 2, where P2GAN achieves highest overall average rank. Detailed MMD metrics are reported in Supplementary. FID scores are reported in Table 3. P2GAN and f-cGAN shows advantages at lower distance values (more overlapping between modes), which suggests that if label matching can be effectively learned (the estimated posteriors are accurate), the discriminator is more powerful.

CIFAR. Results on CIFAR100 [20] are reported in Table 4. Proj-GAN achieves the lowest FID scores, while f-cGAN achieves the highest IS. The adaptive P2GAN model shows competitive high IS and low FID. It is also an interesting confirmation that the P2GAN-w indeed learns a model in between. Results on CIFAR10 are reported in Table 5.

ImageNet. We conduct experiments on ImageNet [36] and compare with all baselines at resolution of 64×64. From Table 4, we see P2GAN models achieve the highest IS and lowest FID. Intra FIDs are computed on 50 classes (samples of these 50 categories are given in Supplementary) due to its high computation cost. Figure 4 shows that for the "flower" class in the last row, Proj-GAN and the proposed P2GAN-w generate diverse samples, while TAC-GAN, f-cGAN and P2GAN all show mode collapse. This can be verified by low max-FID values of the P2GAN-w model. Results of 128 resolution experiments are reported in Table 5.

VGGFace2. VGGFace2 [4] is a large-scale dataset for face recognition, and contains over 9000 identities with around 362 images for each identity. The VGGFace2 dataset is introduced in [6] for image generation task. We follow their experimental setup and construct subsets VGGFace200 and VGGFace500. The images are resized to 64×64. Table 4 shows that the proposed f-cGAN and P2GAN models significantly outperform Proj-GAN and TAC-GAN. To directly measure diversity, we report the min-LPIPS (across all categories) in Table 6. From Figure 3 we can see that Proj-GAN, TAC-GAN and P2GAN all show mode collapse to a certain degree. While P2GAN-w generates sharp and diverse images. Samples from Proj-GAN are also blurry.

4.3. Ablation Studies

A naïve hybrid model could be an ensemble of the Proj-GAN loss and the f-cGAN loss, which requires three parameters v_y, v_y^p and v_y^q. The resulting does not truly untie class embeddings and less elegant. We find it does not show superior performance over dual projection. We provide in Table 7 comparisons of the naïve baseline, an over-parameterization baseline ($\lambda \equiv 0$), and DM-GAN ($\psi \equiv 0$). Also, the training curves on VGGFace200 given in Supplementary shows that over-parameterization alone does not prevent Proj-GAN from failing on this dataset and dual projections indeed benefit from balanced data matching and label matching.

Table 7: FID scores of ablation studies. All models are trained for 60000 iterations on ImageNet, 62000 iterations on CIFAR100, and 50000 iterations on VGGFace200.

	ImageNet	CIFAR100	VGGFace200
Proj-GAN	24.57	10.02	61.27
P2GAN	21.46	10.55	23.15
$\lambda \equiv 0$	23.64	10.82	62.46
$\psi \equiv 0$	26.30	10.95	87.01
Naïve	-	10.58	32.31

5. Conclusion

In this paper, we give insights on projection form of conditional discriminators and propose a new conditional generative adversarial network named Dual Projection GAN (P2GAN). We demonstrate its flexibility in modeling and balancing data matching and label matching. We further rigorously analyze the underlying connections between AC-GAN, TAC-GAN, and Proj-GAN. From the analysis, we first propose f-cGAN, a general framework for learning conditional GAN. We demonstrate the efficacy of our proposed models on various synthetic and real-world datasets.

5.1. Acknowledgments

This research was funded based on partial funding from NSF: IIS 1703883, NSF IUCRC CNS-1747778, CCF-1733843, IIS-1763523, IIS-1849238, and NIH: 1R01HL127661-01 and NIH: R01HL127661-05.

References

[1] Yogesh Balaji, Martin Renqiang Min, Bing Bai, Rama Chellappa, and Hans Peter Graf. Conditional gan with discriminative filter generation for text-to-video synthesis. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 1995–2001. International Joint Conferences on Artificial Intelligence Organization, 2019. 1

[2] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeswar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and R Devon Hjelm. Mine: mutual information neural estimation. *arXiv preprint arXiv:1801.04062*, 2018. 2, 4

[3] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018. 1, 6

[4] Qiong Cao, Li Shen, Weidi Xie, Omkar M Parkhi, and Andrew Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 67–74. IEEE, 2018. 6, 8

[5] 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. 1

[6] Mingming Gong, Yanwu Xu, Chunyuan Li, Kun Zhang, and Kayhan Batmanghelich. Twin auxilary classifiers gan. In *Advances in Neural Information Processing Systems*, pages 1328–1337, 2019. 1, 6, 8

[7] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 1, 3

[8] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(Mar):723–773, 2012. 8

[9] Ligong Han, Ruijiang Gao, Mun Kim, Xin Tao, Bo Liu, and Dimitris Metaxas. Robust conditional gan from uncertainty-aware pairwise comparisons. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 10909–10916, 2020. 1

[10] Ligong Han, Robert F Murphy, and Deva Ramanan. Learning generative models of tissue organization with supervised gans. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 682–690. IEEE, 2018. 1

[11] Ligong Han, Anastasis Stathopoulos, Tao Xue, and Dimitris Metaxas. Unbiased auxiliary classifier gans with mine. *arXiv preprint arXiv:2006.07567*, 2020. 2

[12] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in neural information processing systems*, pages 6626–6637, 2017. 6

[13] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *International conference on machine learning*, pages 1989–1998. PMLR, 2018. 1

[14] 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. 1

[15] Minguk Kang and Jaesik Park. Contragan: Contrastive learning for conditional image generation. 2020. 5, 7

[16] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017. 1

[17] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. In *Proc. NeurIPS*, 2020. 4

[18] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7482–7491, 2018. 2, 4

[19] Murat Kocaoglu, Christopher Snyder, Alexandros G Dimakis, and Sriram Vishwanath. Causalgan: Learning causal implicit generative models with adversarial training. *arXiv preprint arXiv:1709.02023*, 2017. 2

[20] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 5, 8

[21] Chunyuan Li, Hao Liu, Changyou Chen, Yuchen Pu, Liqun Chen, Ricardo Henao, and Lawrence Carin. Alice: Towards understanding adversarial learning for joint distribution matching. In *Advances in Neural Information Processing Systems*, pages 5495–5503, 2017. 4

[22] Yitong Li, Martin Renqiang Min, Dinghan Shen, David E. Carlson, and Lawrence Carin. Video generation from text. In Sheila A. McIlraith and Kilian Q. Weinberger, editors, *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18), New Orleans, Louisiana, USA,*, pages 7065–7072. AAAI Press, 2018. 1

[23] Xuanqing Liu and Cho-Jui Hsieh. Rob-gan: Generator, discriminator, and adversarial attacker. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11234–11243, 2019. 2

[24] Lars Mescheder, Sebastian Nowozin, and Andreas Geiger. Which training methods for gans do actually converge? In *International Conference on Machine Learning (ICML)*, 2018. 4

[25] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014. 1

[26] Chaitanya Mitash, Bowen Wen, Kostas Bekris, and Abdeslam Boularias. Scene-level pose estimation for multiple instances of densely packed objects. In *Conference on Robot Learning*, pages 1133–1145. PMLR, 2020. 1

[27] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative ad-

versarial networks. *arXiv preprint arXiv:1802.05957*, 2018. 4

[28] Takeru Miyato and Masanori Koyama. cgans with projection discriminator. *arXiv preprint arXiv:1802.05637*, 2018. 1, 3, 6

[29] Shakir Mohamed and Balaji Lakshminarayanan. Learning in implicit generative models. *arXiv preprint arXiv:1610.03483*, 2016. 1

[30] Youssef Mroueh*, Igor Melnyk*, Pierre Dognin*, Jerret Ross*, and Tom Sercu*. Improved mutual information estimation, 2020. 4

[31] Anh Nguyen, Jeff Clune, Yoshua Bengio, Alexey Dosovitskiy, and Jason Yosinski. Plug & play generative networks: Conditional iterative generation of images in latent space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4467–4477, 2017. 1

[32] Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-gan: Training generative neural samplers using variational divergence minimization. In *Advances in neural information processing systems*, pages 271–279, 2016. 2, 5

[33] Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2642–2651. JMLR. org, 2017. 1

[34] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems*, pages 8026–8037, 2019. 6

[35] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. *arXiv preprint arXiv:1605.05396*, 2016. 1

[36] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015. 2, 3, 6, 8

[37] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in neural information processing systems*, pages 2234–2242, 2016. 6

[38] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. *arXiv preprint arXiv:1805.08318*, 2018. 1

[39] Han Zhang, Tao Xu, and Hongsheng Li. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5908–5916, 2016. 1

[40] 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. 1