CS230

Image-to-Image Translation with Causal GAN

Xin Jiang
Symbolic Systems Program
Stanford University
xinj6@stanford.edu

Abstract

We propose applying causal GAN [2] to solve image-to-image translation problems. This machinery enables sampling not only from conditional observational distributions but also from interventional distributions. We evaluated causal GAN by comparing it with conditional GAN in both observational and interventional settings, and conclude that causal GAN has better performance in image generation when there are strong causal effects between labels of images. While causal GAN and conditional GAN generate similar results when there is weak or no causal effect between labels.

1 Introduction

Image to image translation is defined as the task of translating one possible representation of a scene into another, given sufficient training data [1]. We propose using a causal generative adversarial networks [2] in image to image translation problems that include strong dependencies between labels. This design of algorithm preserves the causal effect between labels and the causal effect between labels and images, whereas using deep conditional GAN (DCGAN) only captures the dependence between labels and images [3].

The input to this algorithm is a random noise vector N. We then generate a labelLg from a causal controller which controls the distribution the labels will be sampled from when intervened or conditioned on a set of labels. We then use a DCGAN network to output a predicted image that is conditioned on the label."

The work in the paper is important since this is the first exploration of learning a causal implicit generative model to solve image-to-image translation problem. (The first and the only work in causal GAN was used to predict face images (label to image) [2]). This machinery enables sampling not only from conditional observational distributions but also from interventional distributions. With the new design, We can generate images with desired combination of features in it that may not be present in the training set.

2 Related work

We could automate the learning process given a loss function with CNNs, but we still need to design effective losses. This problem could be solved by Generative Adversarial Networks (GANs)[4,5,6,7], which automatically learns a loss function appropriate for satisfying our high-level goal. An extension of GANs is conditional GANs (cGANs), which enables sampling from conditional data distributions by feeding class labels to the generator alongside the noise vectors [3,8]. The cGAN is different in that the loss is learned, and can penalize possible structure that differs between output and target.

CS230: Deep Learning, Winter 2018, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

Conditional GANs could capture the dependence between labels and output images, but could not capture the dependence and causal effect between labels. CausalGAN [2] was introduced and used in face image prediction. Causal GAN captures the dependencies and causal effect between labels given a causal graph.

In this paper, we propose using a causal GAN in image to image translation. This design of algorithm captures the potential causal effect between labels and enables sampling not only from conditional observational distributions but also from interventional distributions.

3 Dataset and Features

We used a dataset of facade images assembled at the Center for Machine Perception [12], which includes 606 (400 for training, 100 for development, 106 for test) rectified images of facades from various sources and which have been manually annotated. Image annotation is a set of rectangles scope with assigned class labels. The basic class labels include facade, window, blind, cornice, sill, door, balcony, deco, molding, pillar, and shop. The specific definition could be find in Appendix 5. A summary of the dataset are shown below in figure 1.

Image & annotation example				
Location	Prague, Czech Republic	Bratislava, Buenos Aires, Frankfurt, Graz, London, Ostrava, Rome, Znojmo	Zurich, Switzerland	Barcelona, Greece, Budapest, USA
Date	2007	2007-2009	2003	2010
Resolution	6 MPx	6 MPx	0.3 MPx	0.6 MPx
Size	213 images	99 images	177 images	177 images

Figure 1. Dataset summary

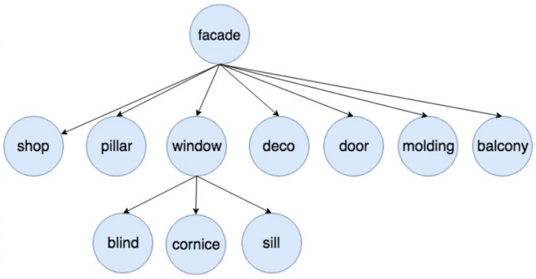

Figure 2.Label causal graph

Based on the definition of the class labels, we constructed a causal graph (figure 2) that represents the causal effect between labels. This causal graph will be used to guide the design of the causal controller.

4 Methods

We adopted the two-step process proposed in [2] for learning a causal implicit generative model over labels and images. 1) The causal controller produce labels that are sampled from observational or interventional distributions. The causal controller is structured based on a given causal graph, and is trained with a Wasserstein GAN [2,11]. 2) We used a DCGAN architecture to generate images based on the labels output from the pretrained causal controller. For the DCGAN, both the generator and the discriminator use modules of the form convolution-BatchNorm-ReLu [9]. Two key features from the prior work is used in this architecture, which are generator with skips and Markovian discriminator (PatchGAN) [1] (see details of generator with skips and Markovian discriminator in Appendix 6). This architecture and the proposed loss functions assures that the generator outputs the label conditioned images[2].

4.1 Causal Implicit Generative Models

Causal implicit generative models provide a way to sample from both observational and interventional distributions [2] (See Appendix 1,2, and 3 for the definition of causality, causal model and intervention).

Prior work observed that in the GAN training framework, generator neural network connections can be arranged to reflect the causal graph structure [2]. For example, given a simple causal graph $X \to Z \leftarrow Y$. Under the causal sufficiency assumption, this model can be written as $X = f_X(E_X), Y = f_Y(E_Y), Z = f_Z(X, Y.E_Z)$, where f_X, f_Y, f_Z are some functions and E_X, E_Y, E_Z are jointly independent variables. This causal graph could be represented by a neural network graph. The forward propagation could be used to represent the functions f_X, f_Y, f_Z. The noise terms (N_X, N_Y, N_Z) can be chosen as independent, corresponding to (E_X, E_Y, E_Z). Then the neural network graph could be used to represent the causal graph $X \to Z \leftarrow Y$.

It has also been approved that given the true causal graph, two causal models that have the same observational distribution have the same interventional distributions for any intervention (Proposition 1) [2].

Proposition 1.
Let $M_1 = (D_1 = (V, E), N_1, F_1, P_{N_1}(.))$,
$M_2 = (D_2 = (V, E), N_2, F_2, Q_{N_2}(.))$ be two causal models, where $P_{N_1}(.)$ and $Q_{N_2}(.)$ are strictly positive densities. If $P_V(.) = Q_V(.)$, then $P_V(.|do(S)) = Q_V(.|do(S))$. where P_v and Q_v stands for the distributions included on the set of variables in V by P_{N_1} and Q_{N_2}, respectively [2].
The consistency between a neural network and a causal graph is defined as follows [2]:

Definition 1.
Let $Z = Z_1, Z_2, ...Z_m$ be a set of mutually independent random variables. A feedforward neural network G that outputs the vector $G(Z) = [G_1(Z), G_2(Z)...G_n(Z)]$ is called **consistent** with a causal graph $D = ([n], E)$, if $\forall i \in [n], \exists$ a set of feedforward layers f_i such that $G_i(Z)$ can be written as $G_i(Z) = f_i(\{G_j(Z)\}_{j \in P_{a_i}}, Z_{S_i})$, where P_{a_i} are the set of parents of i in the causal graph D, and $Z_{S_i} := \{Z_j : i \in S_i\}$ are collections of subsets of Z such that $\{S_i : i \in [n]\}$ is a partition of $[m]$.

Based on the definition of consistency, the causal implicit generative models(CiGM) are defined as follows:[2]:
Definition 2. A feedforward neural network G with output
$G(Z) = [G_1(Z).G_2(Z)...G_n(Z)]$ is called a causal implicit generative model for the causal model $M = (D = ([n], E), N, F, P_N(.))$ if G is consistent with the causal graph D and
$P(G(Z) = X) = P_{[n]}(X) > 0, \forall X$.

4.2 Causal Generative Adversarial Networks

We adopt the method in [2] to train the Causal Implicit Generative Models (CiGMs). First we train a generative model over the labels with causal controller with a Wasserstein GAN, and then train a generative model for the images conditioned on the labels produced by the causal controller with a deep conditional GAN.

1. Causal Controller
   The causal controller is designed for controlling which distributions the images will be sampled from when intervened or conditioned on a set of labels. It inputs a set of independent noise vectors and output binary labels that are sampled from the observational or interventional distribution that satisfies the given causal graph.
   Causal controller is trained with a Wasserstein GAN [2,11]. The gradient term used as a penalty is estimated by evaluating the gradient at points interpolated between the real and fake batches. Therefore, Wasserstein GAN support outputting almost discrete labels, which could be used to train the Causal Controller [2]. We adopted the modified Wasserstein GAN (with penalized gradient) used in [2], which assures 1) convergence in distribution of the Causal Controller output to the discretely supported distribution of labels and 2) given complete causal graph will lead to a nearly perfect implicit causal generator over labels and that Bayesian partially incorrect causal graphs can still give reasonable convergence [2]. The design of the causal controller that samples the distribution that satisfies $window \rightarrow cornice$ is shown in Figure 3.

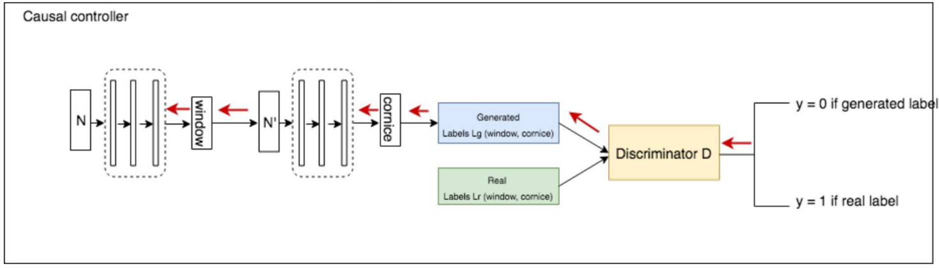

   Figure 3. Design of causal controller
   The generator is a set of fully connected neural networks that are arranged and connected to reflect a predefined causal graph (Figure 3). Similar to DCGANs, the discriminator of

Wasserstein GANs attempts to distinguish between fake and real sample.
The loss for WGANs are:
$L_{wGAN} = E_{x \sim Pg}[G_w(x)] - E_{x \sim Pr}[D_w(x)]$
The objectives for WGANs are:
$G* = argmin_G max_D L_{wGAN}$
Total variation distance (TVD) between the distribution of generator and data distribution was used as a metric to decide the success of the model.
$L_{wGAN}(P, Q) = sup_{A \in F}|P(A) - Q(A)|$, where P and Q are two probability measures on a sigma-algebra F of subsets of the sample space Ω. Intuitively, this is the largest possible difference between the probabilities that the two probability distributions can assign to the same event.

2. Conditional GAN

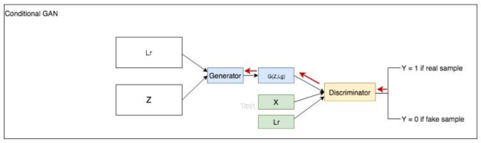

Figure 4. Conditional GAN

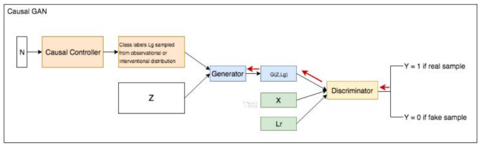

Figure 4. Causal GAN

Conditional GANs learn a mapping from an observed image X and random noise vector Z to Y, $G : \{X, Z\} \rightarrow Y$. It is composed of a generator and a discriminator (Figure 4). The generator models the sampling process through forward propagation given a noise vector and a class label, and get feedback from a discriminator through backward propagation. It attempts to convince the discriminator that the generated sample is from the real distribution by minimizing the opposite of the objective of the discriminator. On the contrary, the discriminator attempts to distinguish between the generated samples from real samples, and should correctly label true sample as 1 and generated sample as 0. Both the generator and the discriminator are trained simultaneously but with opposite objectives. Compared with GANs, conditional GANs input class labels alongside the random noise vector to the generator, and input class labels alongside observed images to discriminator to ensure that the images being generated are conditioned on the class labels.

The Loss of a conditional GAN could be expressed as [3]:
$L_{cGAN}(G, D) = E_{l,y}[logD(y|l)] + E_{l,z}[log(1 - D(G(l, z)|l))]$
where G (generator) attempts to minimize this objective against an adversarial D (discriminator) that attempts to maximize this objective: $argmin_G max_D L_{cGAN}(G, D)$. The loss function of a conditional GAN is different from the loss function of GAN by conditioning on l.
Previous work found it beneficial to mix the GAN objective with L1 or L2 distance[1, 42]. We adopt this design and finalize the objective function as:
$L_{L1}(G) = E_{l,y,z}[||y - G(l, z)||]$,
$L_{cGAN}(G, D) = E_{l,y}[logD(y|l)] + E_{l,z}[log(1 - D(G(l, z)|l))]$,
Objective: $argmin_G max_D L_{cGAN}(G, D) + \lambda L_{L1}(G)$

Combining the causal controller and the conditional GAN, we constructed a causal GAN model as shown in Figure 5. The causal controller samples labels that satisfy an observational or interventional distribution. Instead of feeding general sampled labels to the conditional GAN. Causal GAN feed samples that are sampled based on a causal graph to the conditional GAN.

The work in [2] shows that this architecture and the loss functions without the $L1$ term assures that the generator outputs the label conditioned image distribution, under the assumption that the joint probability distribution over the labels is strictly positive (See Theorem 1 in Appendix 4). We assumed that adding the $L1$ term in the loss function won't violate this finding.

5 Experiments and Results

We built a model to translate Architectural labels to photos, with 400 training images from [12]. Data was split into train and test randomly.

To optimize the Causal Controller (Wasserstein GAN), we used the approach proposed from [2]: set the number of critic iterations to 20, the number of layers in the Wasserstein discriminator to 6, the hidden size for critic of discriminator to 15, and the RMSprop learning rate to 0.00008. We trained for 5000 iterations and achieved a total variation distance (TVD) of 0.004.

To optimize the conditional GAN, we followed the approach proposed from [1]: alternate between one gradient descent step on D, then one step on G. The same as [1], we also divided the objective by 2 while optimizing D, which slowed down the rate at which D learned relative to G. We used minibatch SGD and applied the Adam solver, with learning rate 0.0002, and momentum parameters b1 = 0.5, b2 = 0.999. We trained it with batch size 1.

We evaluated causal GAN by comparing it with the same conditional GAN without the causal controller in both observational and interventional settings. Specifically, we trained a causal GAN and a DCGAN, then ran a 2 by 2 factorial study with the following conditions: 1) $window = 1 \rightarrow cornice = 1$, causal GAN, 2)$window = 1 \rightarrow cornice = 1$, DCGAN, 3)$window = 0, cornice = 1$, causal GAN, 4)$window = 0, cornice = 1$, DCGAN. The assumption is that causal GAN has a better performance than DCGAN for interventional distribution, while there is no big difference between DCGAN and causal GAN for observational distribution.

We trained both causal GAN and conditional GAN for 30 epochs (400 iterations per epoch) with mini batch size 1. As shown in Figure 5, causal GAN generated better images in interventional examples, which had never been shown to the network, and it also generated similar quality result in conditional examples, which were present in the training dataset. Althrough 8 images might not be able to show this difference statistically.

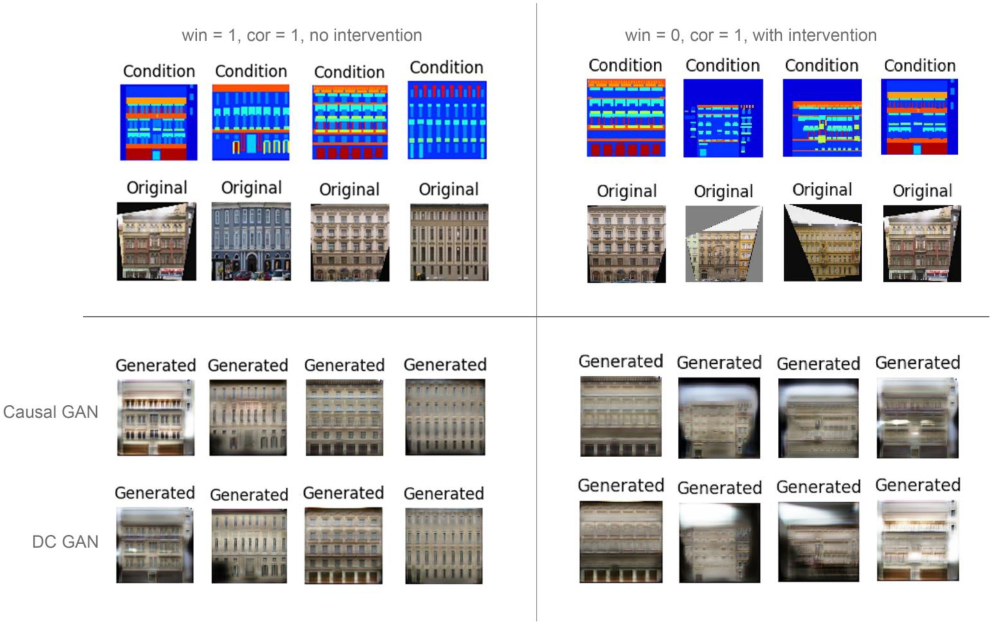

Figure 5. Result for the two models in both settings

6 Conclusion/Future work

In this paper, we explored integrating causality to generative adversarial networks and proposed applying causal GAN [2] to solve image-to-image translation problems. We evaluated causal GAN by comparing it with conditional GAN in both observational and interventional settings, and concluded that causal GAN had better performance in image generation when there were strong causal effects between labels, while generated similar results when there were weak or no causal effects between labels.

In the future, I would want to 1) explore ways to integrate causality to generative model, not only the label space. 2) explore situations when the causal graph is not known, or cannot be completely identified from data, or potentially is influenced by latent factors.

References

[1] Phillip Isola & Jun-Yan Zhu & Tinghui Zhou &Alexei A. Efros (2017) Image-to-Image translation with conditional adversarial networks *arXiv preprint arXiv:1611.07004v2*

[2] Murat Kocaoglu &Christopher Snyder &Alexandros G. Dimakis &Sriram Vishwanath (2018) CausalGAN: Learning causal implicit generative models with adversarial training *arXiv preprint arXiv:1709.02023v2*

[3] Mehdi Mirza & Simon Osindero (2014) Conditional generative adversarial nets *arXiv preprint arXiv:arXiv:1411.1784v1*

[4] I. Goodfellow &J. Pouget-Abadie &M. Mirza &B. Xu &D. Warde-Farley &S. Ozair &A. Courville &Y. Bengio (2014) Generative adversarial nets *NIPS, 2014.*

[5] E.L. Denton &S. Chintala &R. Fergus et al. (2015) Deep generative image models using a Laplacian pyramid of adversarial networks. *NIPS* Pages 1486-1494.

[6] A. Radford & L.Metz & S.Chintala (2015) Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*

[7] T.Salimans &I.Goodfellow &W.Zaremba &V.Cheung &A.Rad- ford &X. Chen (2016) Improved techniques for training GANs. *arXiv preprint arXiv:1606.03498*

[8]Augustus Odena & Christopher Olah & Jonathon Shlens (2016) Conditional image synthesis with auxiliary classifier GANs. *arXiv preprint arXiv: 1610.09585*

[9]S. Ioffe & C. Szegedy (2015) Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv: arXiv:1502.03167v3*

[10] G. E. Hinton &R. Salakhutdinov (2006) Reducing the dimensionality of data with neural networks. *Science, 313(5786):504–507*

[11] Martin Arjovsky &Soumith Chintala &Leon Bottou (2017) Wasserstein GAN. *arXiv preprint arXiv:1701.07875*

[12] Radim Tylecek (2013) The CMP Facade Database *Center for machine perception* http://cmp.felk.cvut.cz/ tylecr1/facade/

7 Appendix

1. Causality
   We could use Pearl's framework (Pearl, 2009) to define causality. It uses structural equations and directed acyclic graphs between random variables to represent a causal model. Consider two random variables X,Y, under the causal sufficiency assumption[3], X causes Y means there exists a function f and some unobserved random variable (exogenous) E, independent from X, such that the value of Y is determined based on the values of X and E through the function f, i.e. $Y = f(X, E)$. The causal graph that represents this relation is $X \rightarrow Y$.
2. Causal model
   A structural causal model is a tuple $M = (V, \epsilon, F, P_\epsilon(.))$ that contains a set of functions $F = \{f_1, f_2...f_n\}$, a set of variables $V = \{X_1, X_2, ...X_n\}$, a set of exogenous random variables $\epsilon = \{E_1, E_2...E_n\}$, and a product probability distribution over the exogenous variables P_ϵ. The set of observable V has a joint distribution implied by the distribution of E, and the functional relations F.
3. Intervention
   An intervention is an operation that changes the underlying causal mechanism and the corresponding causal graph. An intervention is denoted as $do(X_i = x_i)$, which removes the connections of node X_i to its parents. The joint distribution over the variables after an intervention can be calculated in the following way: Since the causal graph is a Baysian network for the joint distribution, the observational distribution can be factorized as $P(x_1, x_2...x_n)$ = $\prod_{i\in[n]} P(x_i|Pa_i)$, where the nodes in Pa_i are assigned to the corresponding values in $\{x_i\}$ where $i \in [n]$. After an intervention on a set of nodes $X_S := \{X_i\}_{i\in S}$,i.e. $do(X_S = s)$, the post-interventional distribution is given by $\prod_{i\in[n]\setminus S} P(x_i|Pa_i^S)$, where Pa_i^S represents the following assignment: $X_j = x_j for X_j \in Pa_i$ if $j \notin S$ and $X_j = s(j) if j \in S$.
4. Theorem 1. Theorem 1.$Assume P_g(x) = P_r(x)$, where g represents generated, r represents real, and x represents label. Then the global minimum of the virtual training criterion C(G) is achieved if and only if $P_g(x, y) = P_r(x, y)$. That is to say, if and only if given a label x, generator output $G(z, x)$ has the same distribution as the class conditional image distribution $P_r(y|x)$.
5. Definition of class labels

(a) facade: bounding box for a single plane wall, from pavement to roof, only complete facades are labeled, as if there is no occlusion by cars or others.

(b) window: entire glass area including borders, subtypes according to subdivision of window panes; all visible windows are annotated even if not within Facade.

(c) blind: any functional obstacle to light on the window, both open or closed.

(d) cornice: decorative (raised) panel above the window.

(e) sill: decorative (raised) panel or stripe under the window.

(f) door: entrance

(g) balcony: including railing, overlap with window when glass is visible behind.

(h) deco: any bigger piece of original art, paintings, reliefs, statues, when no other class is applicable.

(i) molding: horizontal decorative stripe across the facade, possibly with a repetitive texture pattern.

(j) pillar: vertical decorative stripe across the facade, possibly with a repetitive texture pattern, terminators (cap, base) are labeled separately.

(k) shop: shop windows, commercials, signs.

6. Generator with skips and Markovian discriminator Different from an encoder-decoder network [10], Generator with skips add skip connections between each layer i and layer $n - i$, where n is the total number of layers. Each skip connection simply concatenates all channels at layer i with those at layer $n - i$, which reserves and pass all the information in layer i to layer $n - i$. Markovian discriminator restricts the GAN discriminator to only model high-frequency structure, and rely on the L1 term to force low-frequency correctness given that L1 loss could accurately capture the low frequencies[1]. It only penalizes structure at the scale of patches and tries to classify if each $N * N$ patch in an image is real or fake. This Markovian discriminator was run across the images and an averaged response was provided as the ultimate output of D [1].