GAN Dissection: Visualizing and Understanding Generative Adversarial Networks(Critical Review)

David Bau[1]
davidbau@csail.mit.edu
Jun-Yan Zhu[1]
junyanz@csail.mit.edu
Joshua B. Tenenbaum[1]
jbt@csail.mit.edu
William T. Freeman[1]
billf@csail.mit.edu
Hendrik Strobelt[2]
hendrik.strobelt@ibm.com
Bolei Zhou[3]
bzhou@ie.cuhk.edu.hk

[1] Massachusetts Institute of Technology, Boston
[2] IBM Research, Cambridge MA
[3] The Chinese University of Hong Kong

1 Introduction

This paper gives us a look under the hood to see what kinds of things are being learned by GAN(Generative Adversarial Networks) units, and how manipulating those units can affect the generated images.This paper gives us an fascinating look of what happens inside the GAN's.

2 Summary

Given a trained segmentation model (i.e., a model that can map pixels in an image to one of a set of predefined object classes), we can dissect the intermediate layers of the GAN to identify the level of agreement between individual units and each object class. The segmentation model used in the paper was trained on the ADE20K scene dataset and can segment an input image into 336 object classes, 29 parts of large objects, and 25 materials. Dissection can reveal units that correlate with the appearance of objects of certain classes. Two different types of intervention help us to understand this relationship. First, we can ablate those units (switch them off) and see if the correlated objects disappear from an image in which they were previously present. Second, we can force the units on and see if the correlated objects appear in an image in which they were previously absent. Figure 1 provides an excellent overview. For dissection they took an upsampled and thresholded feature map of a unit and compared it to the segmentation map of a given object class. The extent of agreement is

Figure 1: we can see (a) a set of generated images of churches and (b) the results of dissection identifying GAN units matching trees. When we ablate those units (c) the trees largely disappear, and when we deliberately activate them (d) trees reappear. The same insights can be used for human-guided model improvements. Here we see generated images with artifacts (f). If we identify the GAN units that cause those artifacts (e) and ablate them, we can remove unwanted artifacts from generated images (g). Characterizing units by dissection

captured using an IoU (intersection-over-union) measure. They took the intersection of the thresholded image and the pixels defined as belonging to the segment class, and divided it by their union. The result shows what fraction of the combined pixels is correlated with the class. The following examples show units with high IoU scores for the classes table and sofa.

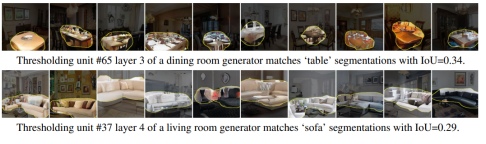

Figure 2: Visualizing the activations of individual units in two GANs

To find causal relationships through intervention, We can say that a given hidden unit causes the generation of object(s) of a given class if ablating that unit causes the object to disappear and activating it causes the object to appear. Averaging effects over all locations and images provides the average causal effect of a unit on the generation of a given class. This set is found by optimizing an objective that looks for a maximum class difference between images with partial ablation and images with partial insertion, using a parameter that controls the contribution of each unit. Here you can see the effects of increasing larger sets of hidden units, in this case identified as being associated with the class tree.

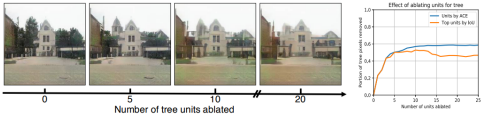

Figure 3: Ablating successively larger sets of tree-causal units from a GAN

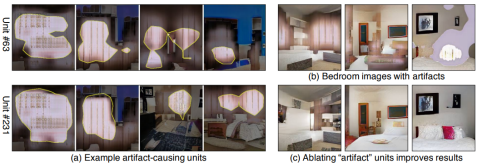

Figure 4: (a) We show two example units that are responsible for visual artifacts in GAN results. There are 20 units in total. By ablating these units, we can fix the artifacts in (b) and significantly improve the visual quality as shown in (c)

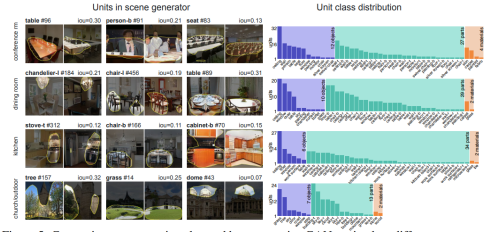

Figure 5: Comparing representations learned by progressive GANs trained on different scene types

3 Review

The units emerge that correlate with instances of an object class, with diverse visual appearances. The units are learning abstractions. The set of all object classes matched by units of a GAN provides a map of what a GAN has learned about the data. The units that emerge are object classes appropriate to the scene type, for example, when they examine a GAN trained on kitchen scenes, we find units that match stoves, cabinets, and the legs of tall kitchen stools. Figure 3 shows this in an elaborating manner . Another striking phenomenon is that many units represent parts of objects: for example, the conference room GAN contains separate units for the body and head of a person.The type of information represented changes from layer to layer. Early layers remain entangled; middle layers have many units matching semantic objects and object parts; and later layers have units matching pixel patterns such as materials, edges, and colors. Here is an interesting layer-by-layer breakdown of a progressive GAN trained to generate LSUN living room images. Compared to a baseline progressive GAN, adding minibatch stddev statistics increases the realism of the outputs. The unit analysis shows that it also increases the diversity of the concepts represented by units. Turning off (ablating) (Figure 4) units identified as associated with common object classes causes the corresponding objects to mostly disappear from the generated scenes. Not every object can be erased, though. Sometimes the object seems to be integral to the scene. For example, when generating conference rooms, the size and density of tables and chairs can be reduced but they cannot be eliminated entirely. By forcing units on, we can try to insert objects into scenes. For example, activating the same door units across a variety of scenes causes doors to appear—but the actual appearance of the door will vary in accordance with the surrounding scene.We also observe that doors cannot be added in most locations. The locations where a door can be added are highlighted by a yellow box, it is not possible to trigger a door in the sky or on trees. Interventions provide insight into how a GAN enforces relationships between objects. Even if we try to add a door in layer 4, that choice can be vetoed later if

the object is not appropriate for the context. By carefully examining representation units, they have found many parts of GAN representations can be interpreted, not only as signals that correlate with object concepts but as variables that have a causal effect on the synthesis of objects in the output. These interpretable effects can be used to compare, debug, modify, and reason about a GAN model.
$\widehat{\text{W}}$ith all this good facts, this work does not give explanation for how object insertion at a place where it can actually be?. For example, a door not be inserted in sky. Also it arises to the question that, how does the GAN supress the signal in the later layers. Understanding these layers of GAN would be next hurdle.

4 Conclusion

The units emerge that correlate with instances of an object class, with diverse visual appearances. Units are learning abstractions. The set of all object classes matched by units of a GAN provides a map of what a GAN has learned about the data. But along with these facts, some questions mentioned previously are still unsolved and are open for research.