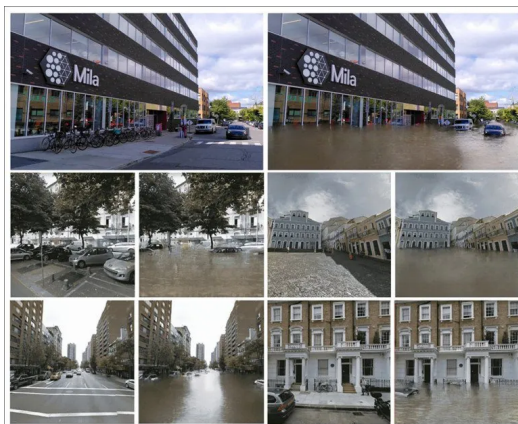


Projecting Climate Change Into Photos With Generative Adversarial Networks

Originally published at <https://www.unite.ai/projecting-climate-change-into-photos-with-generative-adversarial-networks/>



A team of researchers from Canada and the US has developed a machine learning method to superimpose the catastrophic effects of climate change into real photos using Generative Adversarial Networks (GANs), with the aim of reducing 'distancing' – our [inability to relate](#) to hypothetical or abstract scenarios regarding climate change.



ClimateGAN estimates geometry from a calculated depth-map before adding reflectivity in a superimposed water surface.

Source: <https://arxiv.org/pdf/2110.02871.pdf>

The [project](#), titled *ClimateGAN*, is part of a wider research effort to develop interactive environments where users can explore projected worlds that have been affected by floods, extreme heat, and other serious consequences of climate change.

Discussing the motivation behind the initiative, the researchers state:

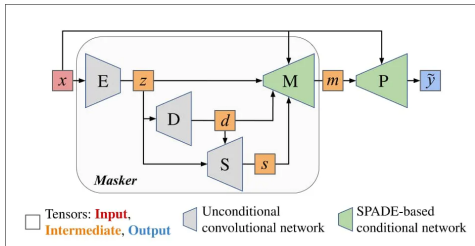
'Climate change is a major threat to humanity, and the actions required to prevent its catastrophic consequences include changes in both policy-making and individual behaviour. However, taking action requires understanding the effects of climate change, even though they may seem abstract and distant.'

'Projecting the potential consequences of extreme climate events such as flooding in familiar places can help make the abstract impacts of climate change more concrete and encourage action.'

A core aim of the initiative is to enable a system where a user can enter their address (or any address) and see a climate-change affected version of the corresponding image from Google Street View. However, the transformation algorithms behind ClimateGAN require some estimated knowledge of height for items in the photo, which is not included in the metadata Google provides for Street View, and so obtaining such an estimation algorithmically remains an ongoing challenge.

Data and Architecture

ClimateGAN utilizes an unsupervised image-to-image translation pipeline with two phases: a Masker layer, which estimates where a level water surface would theoretically exist in the target image; and a Painter module to realistically render water within the boundaries of the established mask, and takes into account reflectivity of the remaining non-obscured geometry above the waterline.



The architecture for ClimateGAN. Input proceeds through a shared encoder into a three-stage masking process before being passed to the Painter module. The two networks are trained independently, and only operate in tandem during the generation of new images.

Most of the training data was chosen from the [CityScapes](#) and [Mapillary](#) datasets. However, since existing data for flood imagery is relatively scarce, the researchers combined existing available datasets with a novel 'virtual world' developed with the Unity3D game engine.

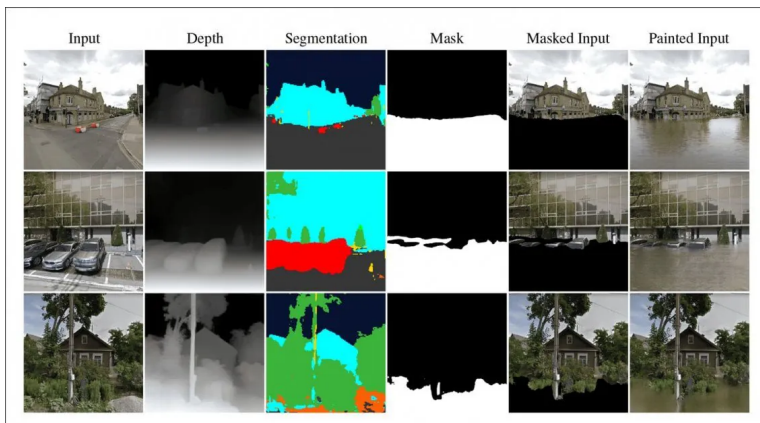


Scenes from the Unity3D virtual environment.

The Unity3D world contains around 1.5km of terrain, and includes urban, suburban and rural areas, which the researchers 'flooded'. This enabled the generation of 'before' and 'after' images for additional ground truth for the ClimateGAN framework.

The Masker unit adapts the 2018 [ADVENT](#) code for training, adding additional data in line with 2019 findings from the French research initiative [DADA](#). The researchers also added a segmentation decoder to feed the Masker unit additional information regarding the semantics of the input image (i.e. labeled information that denotes a domain, such as 'building').

The Flood Mask Decoder calculates a feasible waterline, and is powered by NVIDIA's  **NVDA -4.57%** hugely popular [SPADE](#) in-painting framework.



Click to enlarge. Together with semantic segmentation (third column), depth map information enables delineation of the geometry in a photo, providing a guideline 'flood water'. This can be inferred through machine learning processes, though such information is increasingly being included in consumer-level mobile device sensors. In the second row, we see that the ClimateGAN architecture has successfully rendered a 'flooded' version of the original photo even though the intermediate stages have failed to capture the geometry of a complex scene.

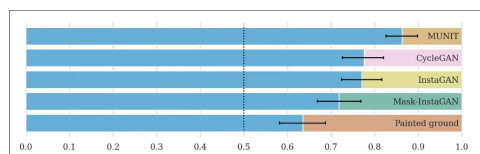
Though the researchers used NVIDIA [GauGAN](#), powered by SPADE, for the Painter module, it was necessary to condition GauGAN on the output of the Masker, and not on a generalized semantic segmentation map, as occurs in normal use, since the images had to be transformed in line with the waterline delineations, rather than being subject to broad, general transformations.

Evaluating Quality

Metrics for evaluating the quality of the resulting images were facilitated by labeling a test set of 180 Google Street View images of varying types, including urban scenes and more rural images from a diversity of geographical locations. The images were manually labeled as *cannot-be-flooded*, *must-be-flooded*, and *may-be-flooded*.



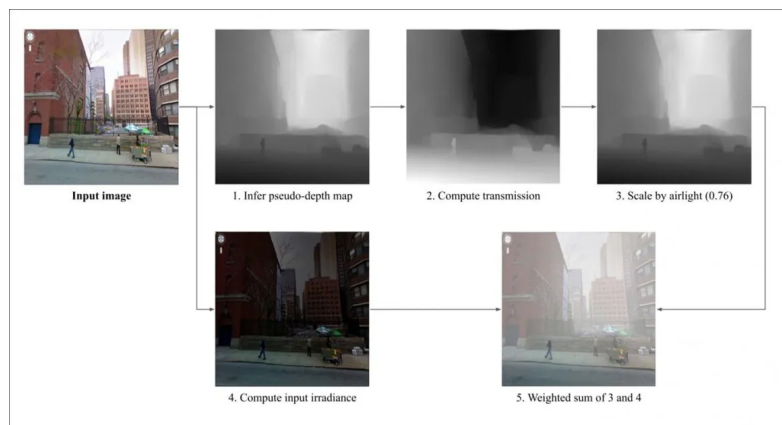
This allowed the formulation of three metrics: *error rate* (perceived prediction areas by size in the transformed image), *F05 Score*, and *edge coherence*. For comparison, the researchers tested the data on prior image-to-image translation (IIT) models, including [InstaGAN](#), [CycleGAN](#), and [MUNIT](#).



In user tests, ClimateGAN was found to achieve a higher degree of realism than five competing IIT architectures. Blue represents the degree to which users preferred ClimateGAN to the studied alternative method.

The researchers concede that the lack of height data in source imagery makes it difficult to arbitrarily impose waterline heights in images, if the user would like to dial up the 'Roland Emmerich factor' a little. They also concede that the flood effects are overly limited to the flood area, and intend to investigate methods by which multiple levels of flooding (i.e. after recession of an initial deluge) could be added to the methodology.

ClimateGAN's code has been [made available at GitHub](#), together with additional examples of rendered images.



In another example, from the GitHub presence for the project, smog is added to a city picture in a way that will be familiar to most VFX practitioners – the depth map receding 'white-out mask', so that the density of smog/fog increases across the distance covered in the photo. Source: <https://github.com/cc-ai/climategan>

Writer on machine learning, domain specialist in human image synthesis. Former head of research content at Metaphysic.ai, until its dissolution into DNEG's Brahma.ai.

Portfolio site: martinanderson.ai

Contact: martin@martinanderson.ai