Whole Earth Fraternal Twin Content for Flight Simulation

Daniel J. Lowe
Collins Aerospace
Salt Lake City, UT
daniel.lowe@collins.com

ABSTRACT

The flight simulation industry desires less expensive and more realistic content for training pilots. Advances in machine learning techniques have brought us closer to creating digital twins of the whole Earth by extracting 3D features from carefully selected photographic imagery. Unfortunately, these databases are too big to distribute, are limited to the texel resolution of the source imagery, and require lots of effort to process and update. Instead, using whole Earth geo-specific metadata, Machine Learning techniques can be used to generate photo-realistic imagery correlated with 3D typical content created at run-time for the whole Earth at any texel resolution. Database modelers' scope of work is greatly reduced whereby they only provide key landmark 3D models where required. Updating Fraternal Twin metadata is much easier than retrieving and processing new satellite imagery.

This paper explains what a Fraternal Twin is and how it is better suited to serve the simulation industry compared to identical Digital Twins for whole-Earth simulation. This paper examines the machine learning processes and metadata needed to create a photo-realistic Fraternal Twin of the whole-Earth that pilots can use for training more cost-effectively. The result is a super high-resolution photo-realistic geo-typical seasonally-accurate representation of the whole-Earth that can be easily distributed and give pilots-in-training realistic visual content with which they can train.

ABOUT THE AUTHOR

Daniel J. Lowe has worked on multiple avionics projects in the past. For the last 15 years he has been a Principal Software Developer working on advanced visual systems. Mr. Lowe holds a Bachelor's degree in Computer Science from California State University, Pomona, a Software Engineering Graduate Certificate from Stevens University, and holds a private pilot's license.

Whole Earth Fraternal Twin Content for Flight Simulation

Daniel J. Lowe
Collins Aerospace
Salt Lake City, UT
daniel.lowe@collins.com

INTRODUCTION

The flight simulation industry desires less expensive and more realistic content for training pilots. Advances in machine learning techniques, specifically Generative Adversarial Networks (GANs) have given us new tools to conceive new solutions to automate content creation. This paper explains how developing techniques for a Whole Earth database can ultimately result in a product that is less expensive and higher quality than a photographic Digital Twin of the Earth.

First, we will examine why a Whole Earth database is the goal. Then why photographic Digital Twin solutions look attractive but don't provide enough cost savings and practicality for simulation. Finally, and primarily, this paper will describe a novel approach to provide a Whole-Earth database solution that promises to be less expensive and have higher image quality. This paper describes research performed by the author to demonstrate how a photo-realistic Fraternal Twin of the whole Earth would work and what whole Earth metadata is needed.

WHOLE-EARTH DATABASES

Building databases for flight simulation is labor-intensive. The databases consist of the terrain (ground imagery) and the 3D content models (mainly buildings and trees). Source data is expensive. Because of this, customers attempt to limit their costs by building or buying databases for only the minimum number of training areas they need. Training areas are large. Commercial aircraft cruising at 35,000 feet (10 km) can see out over 200 miles (320 km). Databases can use lower-resolution imagery away from airports and higher resolution near airports to reduce costs. Military aircraft, however, fly high and low, making it a requirement to have higher resolution imagery over large areas.

Collecting useful imagery is a complex process involving satellites, orthorectification (processing imagery to correct for various optical distortions), atmospheric correction (removing clouds), shadow selection, and color correction. Creating useful images from the raw data is time-consuming.

Pilot Training

It benefits pilots to be able to train anywhere they may fly. The approach flight path to each airport is different. Pilots need to train in unfamiliar areas to better stress the pilots during training. More location options allow pilots to be trained in unfamiliar locations making training more effective. Training with limited simulation areas makes pilots more familiar with the locations during training, which will result in less stress during the training making the simulation less like a real-world experience. Pilots also benefit from training with new airports or routes, making them more prepared when flying in the real world.

Avoid Piecemeal Database Integration

By not providing a whole-Earth solution, databases will inevitably be pieced together from different sources. This creates the issue of integrating the different resolutions and boundaries between different datasets. Vector data from one dataset needs to align properly with satellite photos from another dataset. When they do not, it is a manual effort to reproject the data. Using a whole-Earth database means no effort is spent on integrating different datasets.

Lower Customer Cost

Customers have invested in their databases and want to utilize them as much as possible. Providing a Whole-Earth Database that performs better and costs less will allow customers to move away from their legacy data. Having a database that covers the Earth means we don't build customized databases for one customer.

Developing a whole-Earth database demands that automated processes and tools be developed to process Whole-Earth datasets as opposed to putting a team of Database Modelers to work on a new database with various input sources for every training area a customer requests. The costs of one-off databases are carried by one customer. The cost of a whole-Earth database can be spread across multiple customers.

DIGITAL TWIN

A "Digital Twin" is the virtual representation of an object or system designed to reflect a physical object. A popular approach for creating a Whole-Earth database is to create a virtual digital twin of the Earth by using satellite photo imagery for all of the land masses and by using machine learning techniques to automate 3D feature extraction: road networks, buildings, trees, etc. (see Figure 1) [4]. The source imagery can be in the order of petabytes of storage. The photo imagery must be specifically selected, taking into account the time of day, cloud obscuration, and angle of view. Training a general-purpose 3D model extraction solution requires a huge dataset appropriately sampled across the globe. Multiple strategies are being implemented that involve data fusion with the satellite imagery that may include LiDAR (which is limited in coverage), multispectral imagery, and GIS data (e.g., Open Street Maps, also limited in coverage) to improve the quality of the building extraction [1] [5]. This allows for real-world content to be virtually represented including 3D content. This process is still labor-intensive, and the end product is still expensive. To support sensor simulation, a solution to identify (or classify) the materials in all the world imagery still needs to be developed. As a manual process, sensor classifying a database is too labor intensive for the whole world and AI automation methods have yet to be developed.

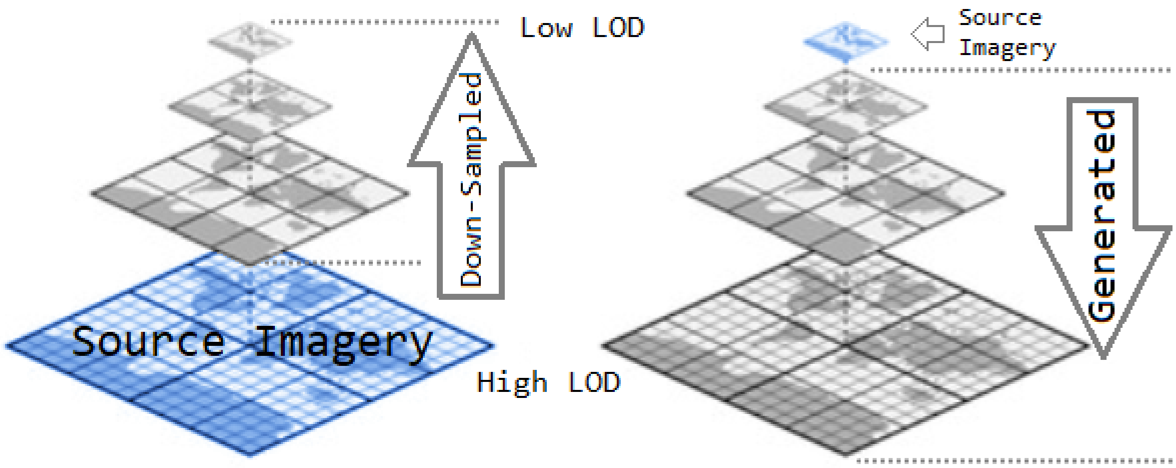
Figure 1 - Microsoft Flight Simulator Digital Twin

Future updates to an area of interest are expensive and will require rerunning the entire pipeline: purchasing new satellite imagery, extracting the 3D objects from the imagery, and storing the data for use by the Image Generators. Using Machine Learning techniques offers promising new ways to build and enhance Digital Twin databases. Still, the problem of the expense related to obtaining, creating, and maintaining the content has not been addressed.

FRATERNAL TWIN

The term "fraternal twin" means biological twins that do not look the same. This term is being used in this paper to describe a technique for generating a specific type of digital twin of the Earth that contains typical content instead of models of the actual content. A Fraternal Digital Twin of the Earth would use Machine Learning techniques to generate photo-realistic terrain imagery at run-time that matches the terrain type, water boundaries, road networks, and building footprints of the real world. Instead of extracting 3D models from the photo, the semantic data (metadata used to distinguish important characteristics of the Earth) already identifies where buildings should be placed and oriented. A template library of buildings and trees would be used to populate the scene that would correlate with the terrain imagery.

A Fraternal Twin database would be much smaller on disk compared to a Digital Twin. A Digital Twin starts with the highest Level Of Detail (LOD) and uses an offline process to produce imagery for lower levels of detail which down-samples the imagery. The Fraternal Twin flips this around by starting with the lowest LOD (~10,000 meters per texture element) and builds the higher LOD (see Figure 2) as needed. There is no limit to how detailed the generated texture image can be - conceivably showing individual blades of grass and grains of sand.



Down Sampled Imagery Generated Imagery (Fraternal Twin)

Figure 2 - Down-sampled Imagery versus Generated Imagery

The lowest LOD of a whole world Fraternal Twin is the base map, at only 200Mb, which has a texture element (texel) resolution of 1,800 meters per texel at an effective perspective of about 13 km (or 42,000 feet). This provides a very organic base-level input into the first level of generated content. The first level of generated content uses a portion of the base map and super-samples the texture going down in elevation by a factor of 2 (so level 1 would be at 6.5 km). Level 2 uses level 1 texture as input to generate super-sampled textures at level 2 (at 3.25 km). The number of networks trained depends upon how close to the ground the eyepoint will get. 16 levels of networks would be needed to generate 0.4-meter texel imagery. Twenty altitude levels of training produce texture at 1 centimeter per texel that would be capable of generating super high-resolution texture for the whole-Earth from cruise altitude to ground level. The imagery is generated on demand and is not stored on disk although the imagery could be cached on disk since training scenarios use the same areas.

Semantic Image Synthesis

Generative Adversarial Networks (GANs), like SPADE, Stable Diffusion, Semantic Diffusion Model (SDM), etc, are a class of machine learning models for general-purpose image generation. GANs among other things offer the ability to take semantic data and turn it into computer-generated photo-realistic imagery. With a model trained to correlate semantic data with a photo, imagery for any part of the Earth can be generated by having the correct representation of shorelines, road networks, buildings, and trees. We trained a SPADE model to generate shoreline imagery from semantic data identifying the water and land. We found that the model was able to generate 512 x 512 terrain tile imagery for any shoreline shape or orientation in under 100 ms making it possible to generate imagery at run-time.

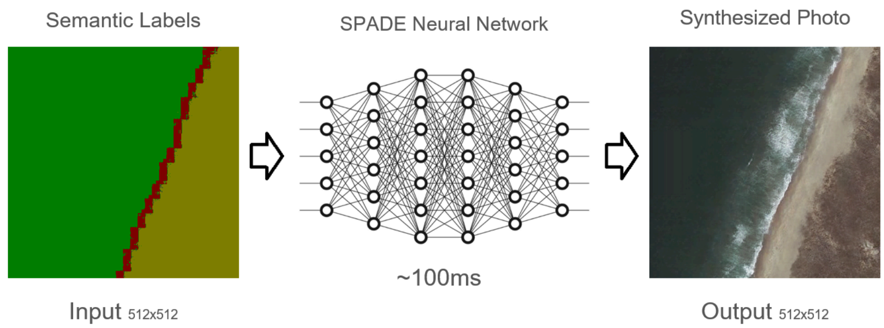


Figure 3 - Using the SPADE Model to create a corresponding image from semantic data

Cost

The cost of generating Simulation Databases is driven by the cost to acquire the data and the effort to process the data to make it ready to be used by the Image Generator. Depending upon the process, future updates to databases can also be cost-prohibitive. There is also the cost of hand-modeling 3D content usually for airports and landmark buildings. To lower the overall cost of content, less data needs to be purchased and less data processing should be needed.

Realistic Content for Simulation

When providing simulation content for training pilots, there is geo-specific content (or landmarks) that pilots expect to be accurately represented in order for pilots to be able to identify and properly orient themselves - airports, specific buildings, major roads, bodies of water, mountains, etc.

Not all content has the same significance to the value of the training scene. Many buildings and roads are not uniquely identifiable by a pilot in flight and as such do not need to be represented with exact models or exact photography. This provides an opportunity to replace all the ground texture and most of the 3D features with geo-typical representations.

More visible from the air than buildings are groups of buildings - shopping malls, industry parks, suburban housing tracts, and downtown skyscrapers. The groups of building types can be represented with geo-typical representations and still give pilots landmarks to navigate with. This also works for most roads and groups of trees.

For landmarks that need to be geo-specific like the Statue of Liberty or the Great Wall of China, there will still need to be a library of carefully selected geo-specific 3D models that would be placed into the scene. The terrain imagery and the 3D models need to be properly correlated. Roads should align with buildings. Images of trees in the terrain should correlate with the location of 3D tree models.

Model Training

Generative Adversarial Networks (GANs) have been steadily improving the quality of imagery created. Starting with Pix2Pix in 2016, Pix2PixHD in 2018 [14], SPADE in 2019 [15], and Stable Diffusion in 2022 [16], AI generation of imagery is now photo-realistic.

Machine learning networks learn how to generate content based on a mapping from input data to output image. Using relatively small training datasets, we were able to train for UK farmlands and CA suburbs using road network semantic data from Open Street Maps (OSM). The accuracy of the road networks was preserved, and photo-realistic imagery was generated to create believable terrain (see Figure 4).

Imagery needs to be collected that represents the variation of imagery that we want to produce. This is not a lot of imagery. A Machine Learning Network for generating terrain imagery is not a general-purpose image generator and does not require the huge datasets that are collected for general-purpose AI image generators like DALL-E2 or Midjourney. Instead, in our research, a dataset of 20 512 x 512 unique images was enough to train for a specific biome and elevation using a SPADE network. The networks were given a road network (or water boundary) to generate imagery as seen in Figure 4. Note that in these images the roads are defined with vector data and the image generated maintains the position of the road network, creating a plausible image for the real-world location.

Figure 4 - Imagery generated from 3 small training datasets - CA beach, UK farmland, CA suburb.

METADATA

Not all coastline, farmlands, and suburbs look the same across the Earth. To train a model to generate imagery over the whole Earth without using training data for the entire Earth, the model must be provided with enough information about the image we want to generate. We will need to use metadata that is available for the entire Earth. The following are seven types of metadata that are readily available that we believe are needed to properly represent the Earth.

Vector Data

Vector Data is used to describe road networks, waterways, building footprints, and trees. At about 100 Mb, Open Street Maps is free to use and is a great source for road networks and water for the whole Earth, but it is not complete for building footprints or trees (see Figure 5). To address this, we can use Machine Learning to train a model to populate the likely locations and orientation of buildings related to the defined roads and water [3]. The same can be done with tree locations. The building footprints help to define the uniqueness of the cities.

We trained a SPADE model to generate building footprints where none existed. A dataset of vector data that had various orientations of roads with buildings was identified. The building footprints were removed leaving only roads to be used as the other dataset. The model was trained to put the building footprints back in. The model was given roads with no buildings, and it positioned building footprints along the road with the proper orientation and position. As the OSM building footprints become more complete, the Fraternal Twin can just use the new data.

Figure 5 - OSM Building Footprint Completeness

Elevation

Digital Elevation Model (DEM) information helps to determine where trees might exist and where snow may be found. Trees do not grow above the treeline. High deserts are different from low-lying deserts. Terrain slope can be derived from elevation grid posts leading to the location of creeks and washes that would not otherwise be identified by OSM vector data. The Machine Learning Model could learn to position trees based on where the water naturally collects leading to a more realistic and organic scene.

Biomes

A biome is a distinct geographical region with a specific climate and vegetation. There are 4 major types of land biomes: grassland, forest, desert, and tundra. This indicates how much moisture an area gets which would translate to how green or brown the terrain should look. Of course, these biomes can be further divided into more subcategories giving us 18 land biomes across the Earth. Image training data would be selected to represent each of the biomes at each elevation we would train for [13]. There are whole Earth biome datasets that are freely available.

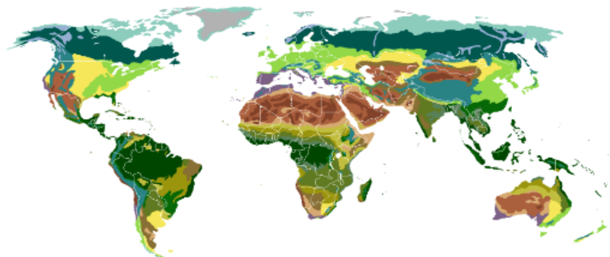


Figure 6 - Land Biomes across the globe

Seasons

For the purposes of flight training, there are typically two seasons that matter for training: summer and winter. For areas that get snow, imagery would need to be collected so that the Fraternal Twin could generate imagery containing snow. The model would learn from the imagery where snow accumulates and what types of roads get plowed and generate terrain imagery accordingly. Much of the Earth's land masses do not see snow and would likely only be trained for the summer season. Figure 7 shows where snow can be found throughout the year. In the southern hemisphere, where the seasons are reversed, snow falls in the months of June to August.

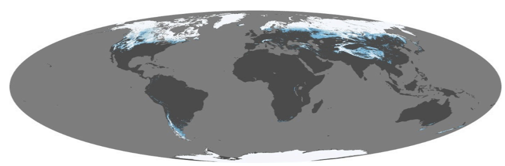


Figure 7 - Snow Cover

Land-Use

Land-use describes how the land has been developed by mankind - industry, urban, farm, untouched, etc. Land-use metadata is available for the entire Earth. Land-use data is not free. Higher-quality land-use data will result in a more accurate simulated world.

Culture/Region

Different cultures build things differently. This type of metadata relates to land-use. In order to render regions properly, imagery from the developed land-use areas is needed from each cultural area that should be properly rendered. The capability to better represent other cultural areas can be improved over time by adding new imagery representing the new cultural areas and then retraining the model. This allows for a whole Earth solution that can mature over time to address customer requirements in a cost-effective way.

Base Imagery

Base Imagery is the lowest-resolution image of the texture tile we are going to generate. We would start selecting a 512 x 512 tile from the image in Figure 8. Each image generated would be used as the base image for the next level. This allows the satellite image of the earth to influence the higher resolution generated imagery.

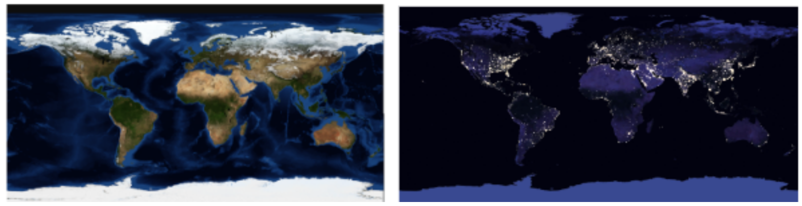


Figure 8 - Earth Day and Night Base Maps

Figure 9 displays the amount of terrain color detail in the Base Map. This helps to provide specific colors around the large desert areas that would otherwise have no varied input to represent the natural color variation of the ground and would not be captured in other metadata.

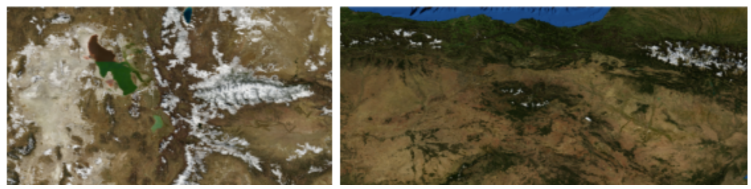


Figure 9 - Color Variations on the Earth

Model Training

The model would use the seven metadata sets and correlate them with the target satellite imagery. Once trained, it would be able to generate the photo-realistic image from the metadata sets only (Figure 10). Once the model learns to generate Iowa farmland it doesn't need any more of that imagery and the Iowa farmland imagery would be generated in other locations with similar metadata. So, we only need to collect enough data to cover all the permutations that the metadata represents. If the metadata changes (new roads are built for example) the model does not need to be retrained as it will already know how to create terrain imagery representing the new road. This makes updating the geo-specific portion of the database cost-effective.

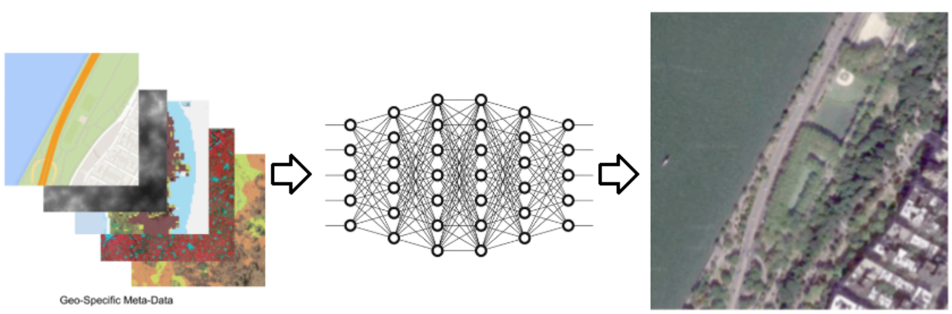


Figure 10 - Model training using Geo-Specific Metadata

Spatial Coherence

Generated content must be consistent (or coherent) across neighboring tile boundaries and between levels of detail (LOD). The content of the generated image must not move around between elevations. The metadata drives the generation of the image, especially the vector data. This will keep roads, buildings, and trees in the correct place, so they line up on tile boundaries and levels of detail.

To demonstrate a solution for neighboring tile coherence we build a SPADE model to generate shoreline imagery. At run-time, vector data was fed to the model to generate the terrain imagery. The Image Generator was flown along the California coastline. Only one elevation level was trained and used to build the scene for all elevation levels. The content was being generated fast enough to fly and the tile boundaries were undetectable for tiles at the same LOD.

To address spatial coherence between levels of detail, using the geographical portion of the lower level of detail imagery as input into the next higher level will keep the image consistent with whatever imagery was generated. This model effectively becomes more like a super-sample image generator providing infinite zoom abilities [7].

Using a SPADE model, we demonstrated that a model could be trained to produce higher levels of detail and maintain the content of the image as higher levels of detail were generated using only the lower LOD (Figure 11). A Fraternal Twin solution would also use other metadata to improve the quality of the image as higher LOD imagery is generated. There are also GANs that produce higher-quality imagery than the SPADE model that was used.

Low LOD Input Generated Higher LOD

Figure 11 - Spatial Coherence - Levels of Detail

Determinism

Content created for flight simulation must be repeatable. The nature of these networks is to generate the same output given the same input. By controlling the input (base map and the metadata) the content generated at each elevation is repeatable. So even though the images are unique for a given area they are the same image that will be generated by different computers.

3D Model Correlation

Generated Imagery will use the vector metadata layer to generate buildings and trees. Since the imagery was generated from the vector metadata layer, it is easy to correlate building and tree positions. Building footprints and trees only get trained and generated at higher levels of detail. At lower levels of detail, the buildings and trees are not represented by 3D models.

For the Fraternal Twin there will need to be a library of 3D models. This library will contain foliage models used across the Earth. Culturally specific sets of buildings can be developed over time as only one set is needed to populate the Earth and there is no end to how large and varied the 3D building library may become. Of course, the more 3D models created increases the cost of the database.

Airports

There is now enough information available to generate realistic looking ground texture that would correlate with the runways, taxiways, tarmacs, and surrounding ground of the airport. By using a Fraternal Twin approach, the airport terrain will blend seamlessly into the scene. Airport vector data is available in Open Street Maps for practically all the world's airports (Figure 12). The Open Street Map airport vector information is likely not accurate enough to be usable for landing airplanes but would suffice for identifying airports from the air. To land the runways, the elevation and vector data would need to be validated or obtained from another certified source.

Geo-typical airports can be made available for the whole Earth and geo-specific airports would likely use the generated terrain imagery with Database Modelers decorating the airport with geographically specific 3D models. Airport content is a big part of flight simulation. Geo-specific airport content will always be a part of aviation simulation databases. However, using Fraternal Twin terrain imagery will decrease the complexity of integrating custom imagery and ultimately lower airport creation costs.

Figure 12 - OSM Metadata on left, Satellite imagery on right

Night

With a Fraternal Twin approach, there is the potential to improve the accuracy of night scenes. Urban lighting is influenced by the orange sodium vapor lamps and light green mercury vapor lamps giving the scene a characteristic orange or green tint (Figure 13) from the pilot's point of view[9] [12]. The same training process can be performed for generating night imagery using the same metadata but instead with night satellite imagery. Training a model to produce a snow night scene based on real imagery will likely produce a more realistic simulation than ever seen before as nighttime illumination on wet/icy streets are hard to compose from photos of daytime satellite imagery.

Southern Italy — Denver, CO — Tokyo, Japan

Figure 13 - Night Satellite Imagery

PERFORMANCE

Performing the inference step of generating terrain imagery with a SPADE-derived model on a standard desktop PC with a consumer graphics card (RTX 2080Ti), 512 x 512 texel terrain tiles were generated in about 400 ms. From our experience, any new ownship reposition will need about 60 terrain tiles to render a 360-degree scene with terrain texture. A reposition near ground level will require another 20 tiles, for a total of 80 terrain tiles, as the texture is generated from the base map that starts high in the sky. This translates to about 30 seconds to generate the imagery for a reposition. This should be fast enough to generate content as the aircraft moves through the scene. The use of higher quality GANs may take longer to generate imagery but the task of terrain tile image generation can also be parallelized as tiles at the same altitude level are not dependent on each other.

After all the Machine Learning Models are trained, content can be generated at run-time for the whole Earth. Metadata can be updated without retraining the Fraternal Twin models so that new roads can be added with minimal effort making it easy to fix metadata issues. For performance reasons, the vector content can be dialed down during run-time to generate less 3D content at run-time and the image generated will match.

SENSORS

Sensor classifying a database requires that the materials in the image be identified so that the Image Generator can simulate how the sensor would 'see' the scene. In the worst case, this is an expensive labor-intensive manual process. To prepare a Whole-Earth Fraternal Twin database to support sensors, only the imagery used in the training datasets (for all altitude levels) needs to be materially classified. The same Fraternal Twin Machine Learning techniques can be used to correlate the metadata with a sensor-classified "image" to then be able to generate material classified imagery that would be correlated with the out-the-window imagery. This greatly reduces the cost of bringing a sensor classified database to market. The set of training data for the Fraternal Twin Whole-Earth is much smaller than the petabytes of satellite imagery needed for a non-Fraternal Digital Twin of the entire Earth. The Machine Learning models would be trained on the material classification and learn to generate high-resolution material classification maps that would correlate with the generated Fraternal Twin imagery.

CONCLUSIONS

Using a Fraternal Twin approach to generate photo-realistic content for aviation simulations will result in purchasing and processing less satellite imagery. Updating Fraternal Twin metadata is much easier than retrieving and processing new satellite imagery, allowing for small area updates and lower costs. All this translates into lower costs for a whole-Earth database. The technology to support this approach is ready to be utilized. The quality and realism of the generated photography are now ready for use in our industry.

This paper has explored what is needed to make a Fraternal Twin work. This paper also shared the research performed to test the technology behind a Fraternal Twin. This paper has shown how a Fraternal Twin can lower the cost of the simulation database and still provide a realistic simulation environment in which pilots will want to fly. The goal of generating a cost-effective, super high-resolution, photo-realistic, geo-typical, seasonally accurate representation of the whole Earth that can be used with sensors and is easy to distribute is at hand.

Future Work

Most of the research referenced in this paper was performed using a SPADE model which has its challenges in generating imagery that looks real. Changing to a Stable Diffusion model [2] [6] [8] should result in the generation of higher-quality imagery.

Most of the research was performed with a SPADE model that only used 3 input channels. The model needs to be adapted to utilize more input channels of the metadata defined in this paper.

This research was only performed on models that generated daytime out-the-window content. Training the model on material-classified data to generate material-classified data from the metadata is needed for a proof of concept.

ACKNOWLEDGEMENTS

I would like to thank my colleagues at Collins Aerospace for their input and expertise with regard to large-area content in simulation. I would like to thank Rishabh Kaushik for helping me get started with the Machine Learning models. I would especially like to thank Larry Seppi who saw the vision and helped me stay the course. I would like to thank Einav Kiperman, Chris Longhurst, and Mollie Mero for reviewing this paper and giving constructive feedback.

REFERENCES

[1] Weijia Li, Conghui He, Jiarui Fang, Juepeng Zheng, Haohuan Fu, and Le Yu. Semantic Segmentation-Based Building Footprint Extraction Using Very High-Resolution Satellite Images and Multi-Source GIS Data. 2019 https://www.mdpi.com/2072-4292/11/4/403

[2] Weilun Wang, Jianmin Bao, Wengang Zhou, Dongdong Chen, Dong Chen, Lu Yuan, Houqiang Li. Semantic Image Synthesis via Diffusion Models. https://arxiv.org/pdf/2207.00050

[3] Javier Garza. (2023). Transforming a Digital World into Real Insights Using Synthetic Data. *The Interservice Industry, Training, Simulation, and Education Conference (I/ITSEC).*

[4] Simon Skinner. (2023). Simulation model abstraction issues for Digital Twins; Separated at Birth? *2023 -The Interservice Industry, Training, Simulation, and Education Conference (I/ITSEC).*

[5] Ido Ariav, David Zaphir, Asaf Avinoam, Alon Faraj, Yisachar Shapira. (2023). Automatic Creation of High Fidelity Open Terrain Digital Twins for Off-Road Autonomous Vehicles Training and Validation., *The Interservice Industry, Training, Simulation, and Education Conference (I/ITSEC).*

[6] Axel Sauer, Frederic Boesel, Tim Dockhorn, Andreas Blattmann, Patrick Esser, Robin Rombach, (2024). Fast High-Resolution Image Synthesis with Latent Adversarial Diffusion Distillation. Retrieved from: 2403.12015 (arxiv.org)

[7] How to make an infinite zoom art with Stable Diffusion. Retrieved from: https://stable-diffusion-art.com/infinite-zoom/

[8] Accelerating Stable Diffusion Inference on Intel CPUs. Retrieved from: https://huggingface.co/blog/stable-diffusion-inference-intel

[9] NASA's Black Marble. Retrieved from: https://blackmarble.gsfc.nasa.gov/

[10] Evans, Cindy, Stefanov, Will. Cities at Night: The View from Space. Retrieved from: https://earthobservatory.nasa.gov/features/CitiesAtNight

[11] Snow Cover. Retrieved from: https://earthobservatory.nasa.gov/global-maps/MOD10C1_M_SNOW

[12] Night Satellite Imagery from: https://eol.jsc.nasa.gov/DatabaseImages/ESC/large/ISS016/ISS016-E-26150.JPG, https://eol.jsc.nasa.gov/DatabaseImages/EFS/highres/ISS016/ISS016-E-27586.JPG

[13] Biomes of the world from: https://ecoregions.appspot.com/

[14] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, Alexei A. Efros. Image-to-Image Translation with Conditional Adversarial Networks. 2018. https://arxiv.org/pdf/1611.07004

[15] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, Jun-Yan Zhu. Semantic Image Synthesis with Spatially-Adaptive Normalization. 2019 https://arxiv.org/pdf/1903.07291

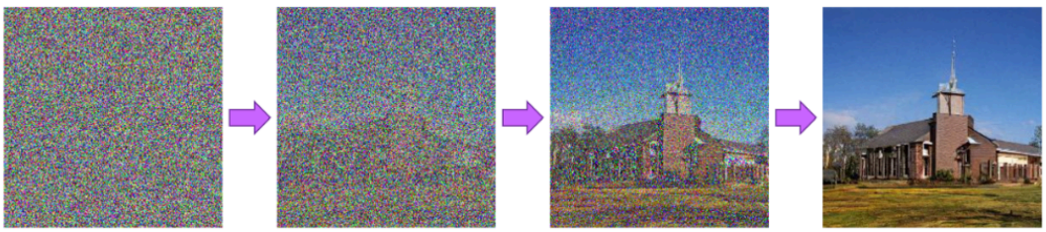
[16] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Muller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, Kyle Lacey, Alex Goodwin, Yannik Marek, Robin Rombach. Scaling Rectified Flow Transformers for High-Resolution Image Synthesis. 2024 https://arxiv.org/pdf/2403.03206

APPENDIX

Semantic Image Synthesis with SPADE

Generated Texture - Yellow Paint on Asphalt

Stable Diffusion



https://colab.research.google.com/github/huggingface/notebooks/blob/main/diffusers/diffusers_intro.ipynb

Stable Diffusion Infinite Zoom In [7]

https://huggingface.co/spaces/ArtGAN/stable-diffusion-2-infinite-zoom-out

Semantic Image Synthesis with Stable Diffusion [2]

Semantics Generated Semantics Generated Semantics Generated Semantics Generated

Semantics Diverse generated images from different noise