CTP Research – Matt Filer (17021871)

A toolkit to produce inexpensive and realistic skyboxes for games

Abstract

This report outlines research undertaken in preparation for a project aiming to achieve the creation of an engine agnostic deep learning driven offline skybox generator for games, to solve the overarching question: "how can pre-computed static skies be brought up to modern standards utilising recent advances in modern science and technology?". The project will focus on collecting a useable dataset, training a machine learning model to reproduce cloud formation and depth data, and producing a scene using volumetric path tracing and existing sky models from that reproduced data.

Introduction and Literature Review

This project was conceived in response to a continued industry interest in realistic static skyboxes for targeting projects at lower end systems. Existing solutions for this are outdated or non-customisable, with a lack of modern alternatives other than expensive realtime renders – the closest available being a tool called Vue (E-on, 2019). A current example of the need for this low budget requirement is virtual reality, as mentioned when interviewing Napper (2019) – VR content production requires far lower graphical budgets due to the higher rendering cost of a standard frame in comparison to non-VR (Wilson, 2015). A realistic skybox precomputed offline would be perfect in this scenario to save resources, particularly as VR experiences are typically short and wouldn't need to show time progression.

Figure 1 - a sky produced in Vue.

The purpose of conducting this research report is to be able to identify suitable methods of data collection for supplying an appropriate machine learning algorithm with the information required to reproduce believable clouds, and to understand the science behind the atmosphere and the makeup of a cloud. These two key pieces of information will form not only a better understanding of the achievability of the project but allow for identification of strengths and weaknesses in past methodologies.

While this project hasn't been exactly attempted before, the paper "Modelling of Clouds from a Single Photograph" (Dobashi, 2010) describes a similar process for cloud data collection as proposed by this project, reportedly with a high degree of accuracy. This method was low cost compared to previous attempts by larger organisations like NASA, who calculated depth data by radiation measurements of the sky. Through research, it appears that the high-cost solutions by NASA were not perfect however, as evaluated in the paper "Determination of the Scaled Optical Thickness of Clouds from Reflected Solar Radiation Measurements" (King, 1987), explored further in the research findings.

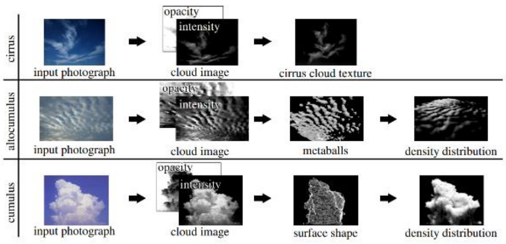

Figure 2 - the process described by Dobashi (2010).

This report aims to answer:

- Is the project feasible, based on previous attempts and recent advancements?
- What is the science behind light scattering, both in clouds and the atmosphere?
- Which machine learning framework should be used?
- What existing sky model is best suited for the project?

Research Methods

The research conducted for this project was that of fact-finding, rather than opinion. For that, primary sources were focussed on, to capture the best data available. In some cases, secondary sources were used to evaluate pros and cons of past methods and available frameworks. These secondary sources should be backed up by co-authors or reputable publishers, to ensure that the thoughts they provide are accurate and of interest.

Due to the lack of prior attempts at a project such as this, it is important that the research is conducted in an unbiased method to enable as truthful an outcome as is possible, as this may form decisions for future advances in this field. To answer the questions posed to the report, a focus was placed on finding papers from a variety of backgrounds: both scientific, industry, and academic. Key data was sourced through work conducted by NASA in the 1970s and 80s in their projects categorising and identifying cloud formations and parameters.

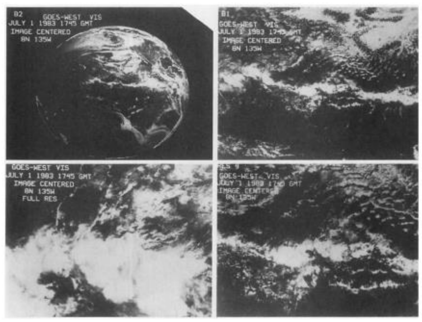

Figure 3 – NASA sample cloud imagery from the 1980s.

Research Findings

Research has shown that game skybox production methods vary by computing era, with early 3D games utilising compressed 360 photos or matte paintings used as textures rendered on a skybox dome (Caspar, 2017). Recent advancements in more modern solutions pull significantly higher amounts of computing resources to calculate skies in realtime from some data pre-computed offline. While these modern effects arguably produce improved visual results, the methods in which they are achieved can cause noticeable performance impacts on lower end systems (Bouthers, 2008). The advantage that these realtime skies provide is the ability for changing weather and time of day effects, however in a range of scenarios (particularly non-open world games) this functionality is negligible, and a realistic static sky would suffice.

Recently, as detailed by Bauer (2019), Red Dead Redemption 2 utilised noise maps for cloud coverage, modified in height and density by a texture look up table (LUT) which was sampled by current in-game weather. Lower quality lighting calculations were performed on clouds further from the player, with some raymarching being performed on closer clouds, linearly interpolated across the clouds surface. In a talk covering the skies of Horizon Zero Dawn, Schneider (2015, 2017) stated that the team went through many iterations before landing on their final solution. Some failed experiments included modulating high poly models using 3D textures baked from Houdini, or even producing cloud models using fluid solvers, although this created a barrier of entry for the art department.

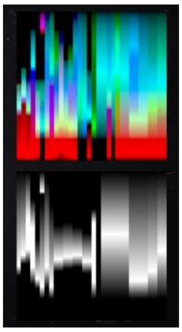

Figure 4 - RDR2's height LUT.

At the absolute highest end of computing power, movie studios like Pixar and Disney have developed their own rendering technologies for clouds, with Schneider (2011) detailing the creation of the clouds for the movie Rio. This movie utilised SmogVox, an internally developed raytracing renderer, to render volumetric clouds produced through Houdini. The issue with these solutions is that frames can take minutes or even hours to render as a result of the computing complexity, clearly not ideal in a game where frame time targets are in the milliseconds. As explained by Thacker (2018), Disney released a dataset for a single cloud produced for Moana which is sampled from a source image, ending up over 2.8GB in size.

Figure 5 - a cloud rendered in SmogVox, with data from Houdini.

The foundation of a skybox comes from the background light, and there are a number of ways to produce this correctly utilising existing understandings behind light scattering in the atmosphere. As light from the sun enters the Earth's atmosphere it is scattered by small particles and aerosols (Nishita, 1998). This phenomenon can be calculated using the Rayleigh and Mie scattering patterns respectively. A few models exist to create atmospheric simulations that reflect this, including the Hosek-Wilkie model (2013), and the Preetham model (1999) – each requiring sun elevation, sky turbidity, and ground albedo as inputs (Scratchapixel, 2015). In a comparison of the two, Kol (2012) concluded that Hosek-Wilkie produces unsatisfactory results for certain values of turbidity and albedo, although both Preetham and Hosek-Wilkie models do suffer from numerical instability which could be undesired for realtime uses in games. This however is not an issue for the offline use required by the project.

Figure 6 - Hosek-Wilkie (left), vs Preetham (right).

The next component of a skybox is clouds: light enters a cloud from the atmosphere and bounces between the water droplets within it, causing dispersion of the light (Satilmis, 2016). This produces bright and dark spots on the cloud relative to the its density and depth in each region (Bouthors, 2008). Volumetric path tracing can be utilised in a renderer to reproduce this effect mathematically, tracing paths that light would bounce throughout a voxelised cloud object, and brightening those areas appropriately. As mentioned previously, linear interpolation can be performed to reduce the number of calculations required to light a cloud's entire surface (Wrenninge, 2015). As well as light, there are varying types of clouds and heights that they can lie in the atmosphere.

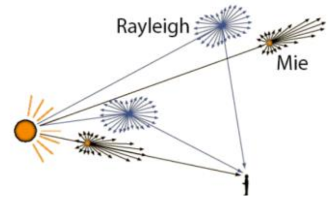

Figure 7 - the effects of Rayleigh and Mie scattering.

Three main shape categories are observed: cumulo, strato, and cirro; as well as three main height categories: above 20,000 ft, 6,500 – 20,000 ft, and below 6,500 ft. Higher level clouds are typically more dispersed, whereas lower level clouds will be more defined in shape (Nelson, 2013).

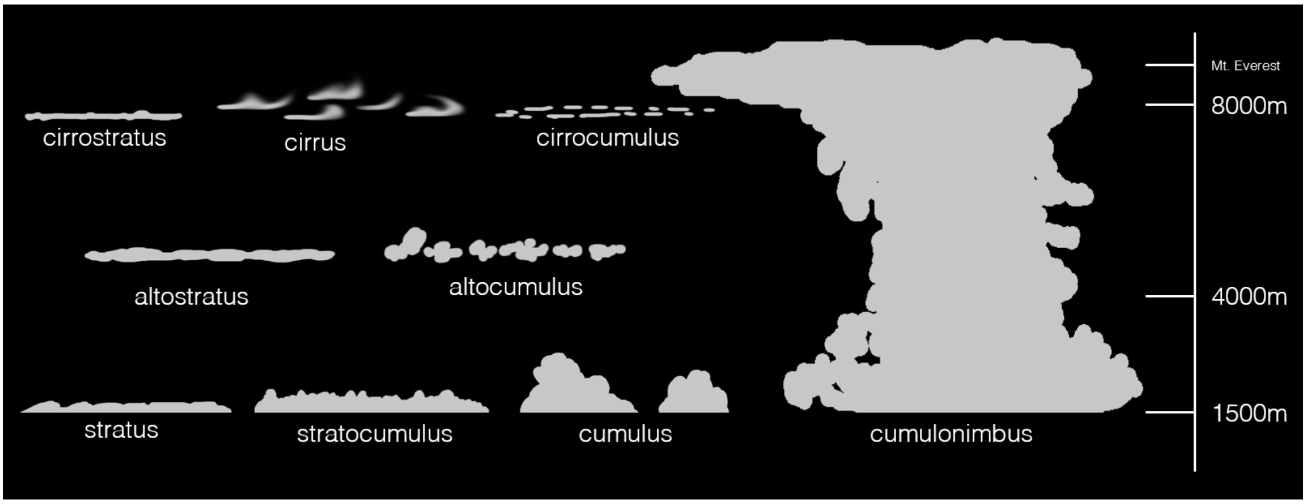

Figure 8 - examples of various cloud types, and the height at which you may find them.

Cloud depth data has been captured in the past using a variety of methods, such as the measurement of solar radiation. Using AVHRR data, differences between spots of radiation in a cloud can be utilised to calculate how much has been absorbed, and therefore, how dense it is (Grainger, 1990). Other cheaper solutions such as measuring UV reflectiveness (Ahmad, 1982), or infrared reflectiveness (Curran, 1981) have been trialled by NASA, although as evaluated by King (1987), 'in none of these cases was it possible to verify the optical thickness [...] by in situ intercomparison'. A more recent approach by Dobashi (2010) explored separating clouds from an input image by interpolating atmospheric colour across neighbouring pixels and removing that colour from each pixel. With the separated cloud, its depth could then be predicted using the remaining opacity and light intensity. From examples given by the author, this worked well and reliably allowed data to be extracted from images for cirrus, altocumulus, and cumulus clouds - this result may have bias, however.

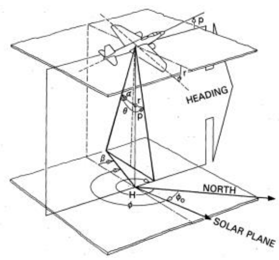

Figure 9 - diagram of a NASA observation craft.

A few alternate methods to detect clouds in images exist, including tests carried out by Lee (1990) where high resolution data was passed through an early neural network which could guess cloud types based on shape to a high level of accuracy. More recently, Jeppesen (2019) released the Remote Sensing Network, which utilised newer deep learning methods to achieve a similar result with a higher degree of accuracy.

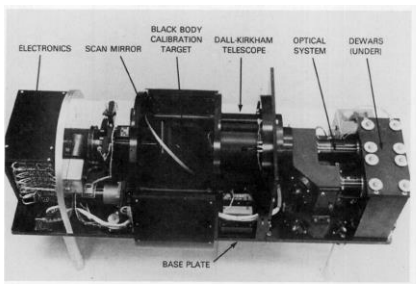

Figure 10 - NASA's radiation scanner, used to identify clouds.

As this project intends to use machine learning to produce its clouds, potential libraries were explored to find the most appropriate. While a wide range of libraries exist such as Keras and Theano, the two most popular are PyTorch and Tensorflow. PyTorch is a Python implementation of the LUA-based Torch framework created by Facebook, while Tensorflow (previously DistBelief) is a Python framework created by Google (Dubovikov, 2017). Both libraries have their own advantages and disadvantages, with the key difference between the two being that Tensorflow is built to use a static graph, while PyTorch uses a dynamic graph – this makes PyTorch great for rapid development, but Tensorflow more suited to permanent bespoke models. There is also considerably more community support for Tensorflow, as PyTorch is in an earlier stage of development (Jain, 2018).

To collect a large dataset, Google Streetview is intended to be utilised. To pull as much data from the Streetview images, high dynamic range (HDR) was explored. HDR is a technology that captures images using 32-bits for each channel in each pixel, allowing for floating point numbers, instead of the standard low dynamic range (LDR) image format, which allows only 8-bits per channel (Khanna, 2019). Although HDR results in a larger filesize, the data is far more accurate, with a useable luminance value which can allow for realistic exposure alterations on a captured image. A format to represent this HDR data is OpenEXR (Industrial Light and Magic, 2000). While using Streetview for the primary data source would provide only LDR images, in recent years the process of converting LDR to HDR using machine learning has been explored. Training on a variety of HDR and LDR versions of the same capture allows the model to understand the differences between the formats and predict the conversion for any other given LDR source image (Marnerides, 2018). Utilising HDR conversion for the Streetview images will allow access to the best luminance data possible, useful for sampling cloud depth data based on density from subsurface scattering.

Figure 11 - a spherical photo from Google Streetview.

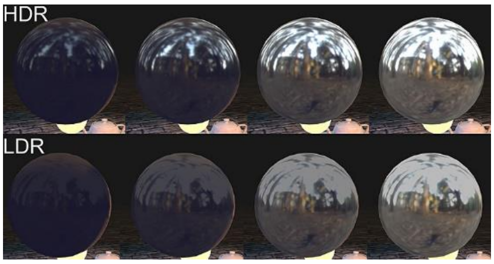

Figure 12 - exposure shown at varying levels across HDR and LDR captures of the same scene.

Conclusion and Recommendations

Based on the research conducted, is recommended that firstly the project will use Tensorflow to build its machine learning model as it is favoured for bespoke solutions. Data will be collected for the machine learning model from Google Streetview due to it being such a large free dataset, and attempted to be converted to HDR using new LDR conversion methods. Cloud depth data will be pulled from these HDR images if successful, using a method similar to that pioneered by Dobashi (2010), however the Hosek-Wilkie sky model will be used as a background to identify cloud pixels rather than a guess from nearby pixel data. If these primary methods create unforeseen limitations or issues, secondary methods will be trialled and reviewed in the final report.

References

- Ahmad, Z *et al.* (1983, p.29-34) *Advances in Space Research.* Volume 2. Amsterdam, Netherlands.
- Bauer, F (2019) *Creating the atmospheric world of Red Dead Redemption 2* [SIGGRAPH 2019], Los Angeles. 28 July.
- Bouthors, A (2008) *Real-time realistic rendering of clouds.* Thesis, Grenoble University.
- Bouthors, A *et al.* (2008) *Interactive multiple anisotropic scattering in clouds.* [ACM SIGGRAPH (I3D) 2008], California. 15 February.
- Caspar, L (2017, p.23-40) *Drawing: Research, Theory, Practice.* Volume 2, Ulster University, Northern Ireland.
- Curran, R *et al.* (1981, p.635-647) *Journal of the Atmospheric Sciences.* Volume 39. NASA Goddard Space Flight Centre, Greenbelt.
- Dobashi, Y *et al.* (2010, p.2083-2090) *Pacific Graphics 2010.* Volume 29. Hokkaido University, Japan.
- Dubovikov, K (2017) PyTorch vs TensorFlow – spotting the difference *Towards Data Science* [blog]. Available from: https://towardsdatascience.com/pytorch-vs-tensorflow-spotting-the-difference-25c75777377b [Accessed 7th December 2019]
- E-on (2012) *Vue* (2019) [computer program]. Available from: https://info.e-onsoftware.com/more-info-vue [Accessed 10th December 2019]
- Fatih, E *et al.* (2017, p.755-758) *2017 International Conference on Computer Science and Engineering*. Volume 1. Antalya, Turkey.
- Grainger, R (1990) *The Calculation of Cloud Parameters From AVHRR Data.* Thesis, University of Auckland.
- Industrial Light and Magic (2000) *OpenEXR* (2019) [computer program]. Available from: https://www.openexr.com [Accessed 7th December 2019]
- Jain, Y (2018) Tensorflow or PyTorch: The force is strong with which one? *Medium* [blog]. Available from: https://medium.com/@UdacityINDIA/tensorflow-or-pytorch-the-force-is-strong-with-which-one-68226bb7dab4 [Accessed 6th December 2019]
- Jeppesen, J *et al.* (2019, p.247-259) *Remote Sensing of Environment.* Volume 229. Aarhus University, Denmark.
- Khanna, M (2019) HDR Imaging: What is an HDR image anyway? *Towards Data Science* [blog]. Available from: https://towardsdatascience.com/hdr-imaging-what-is-an-hdr-image-anyway-bdf05985492c [Accessed 6th December 2019]
- King, M (1987, p.1734-1751) *Journal of the Atmospheric Sciences.* Volume 44. NASA Goddard Space Flight Centre, Greenbelt.
- Kol, T (2012) *Analytical sky simulation.* Thesis, Utrecht University.
- Lee, J *et al.* (1990, p846-855) *IEEE Transactions on Geoscience and Remote Sensing.* Volume 28. South Dakota School of Mines and Technology, USA.
- Marnerides, D *et al.* (2018, section 2, part 1) *EUROGRAPHICS 2018.* Volume 37, Number 2.
- Napper, G (2019) Twitter Messages to Matthew Filer, 10 December. Available from: https://docs.google.com/document/d/1BFEjwzYXc5_E4Ul9GorqjUZNse3mKLDjnOYIrkeUT1M/edit
- Nelson, R (2013) Cloud Types: Identification Basics. *UntamedScience* [blog]. 14 October. Available from: https://untamedscience.com/blog/cloud-types/ [Accessed 24 November 2019]
- Nishita, T (1998, p.1-10) *Rendering Techniques '98.* Volume 1. University of California, USA.
- Preetham, S (1999) *A practical analytical model for daylight.* Research, University of Utah.
- Quilez, I (2010) Dynamic Clouds. *Iquilezles* [blog]. Available from: http://www.iquilezles.org/www/articles/dynclouds/dynclouds.htm [Accessed 27 September 2019]
- Rossow, W (1988, p.201-213) *Journal of Climate.* Volume 2. NASA Institute for Space Studies, New York.
- Rozenberg GV (1974, p.14-24) *USSR Atmosphere and Ocean Physics.* Volume 10. Russia.
- Satilmis, P (2016) *High Fidelity Sky Models.* Thesis, University of Warwick.
- Schneider, A (2011) *Clouds in the skies of Rio* [SIGGRAPH 2011], Vancouver. 7 August.
- Schneider, A (2015) *The realtime volumetric cloudscapes of Horizon Zero Dawn* [SIGGRAPH 2015], Los Angeles. 13 August.
- Schneider, A (2017) *Authoring realtime volumetric cloudscapes with the Decima Engine* [SIGGRAPH 2017], Los Angeles. 30 July.
- *Scratchapixel* (2015) Simulating the Colors of the Sky. *Scratchapixel* [blog]. 10 October. Available from: https://www.scratchapixel.com/lessons/procedural-generation-virtual-worlds/simulating-sky/simulating-colors-of-the-sky [Accessed 23 November 2019]
- Terrell, R (2017) *glsl-atmosphere* [computer program]. Available from: https://github.com/wwwtyro/glsl-atmosphere [Accessed 22 November 2019]
- Thacker, J (2018) Download Disney's data set for Motunui island from Moana. *CG Channel* [blog]. 4 July. Available from: http://www.cgchannel.com/2018/07/download-disneys-data-set-for-motunui-island-from-moana/ [Accessed 28 September 2019]
- Wilkie, A *et al* (2013) *Predicting Sky Dome Appearance on Earth-like Extrasolar Worlds* [SIGGRAPH 2012], Los Angeles. 5 August.
- Wilson, T (2015) *High performance stereo rendering for VR* [talk at San Diego VR Meetup], San Diego. 20 January.
- Wrenninge, M (2015) *Art-Directable Multiple Volumetric Scattering.* Research, Pixar.
- Zhang, J (2017) *Learning High Dynamic Range from Outdoor Panoramas* [ICCV 2017], Venice. 22 October.

Project Log

Date	Tasks completed prior to meeting	Questions arisen in meeting	Tasks to complete for next meeting
26th September	Completed research over the summer into subject area, discovered that volumetrics are a popular modern solution for simulating clouds.	Should the volumetrics be rendered offline or at runtime?	Create a draft proposal, continue research to understand initial approach. Definitively decide on offline or realtime for the project.
9th October	Draft proposal completed and reviewed, further research completed. Decision made that project will focus only on offline skyboxes.	How should the project be approached? Gathering a dataset or building the machine learning model?	Look into methods that can be used to gather a dataset. Streetview was suggested as a potential data source.
20th October	Streetview seems to be a useable data source, but its APIs are paid-for. Did manage to scrape some information for free.	Is it possible to scrape more data from Streetview for free? If not, will free HDR skybox image sites suffice?	Look into HDR skybox sites. Try and scrape more information from Streetview, even if only image data.
1st November	Tool created to rip images from Streetview. No additional data able to be gathered, but images should suffice. HDR skybox sites not a big enough data set.	Can data be pulled from the Streetview images, E.G. ground position, sun position?	Try and implement the ability to cut out the ground from the Streetview images. Look into possibly scraping more metadata.
15th November	Ground able to be cut from Streetview images, although unreliable. More metadata pulled from Streetview, including neighbouring image IDs to allow for recursion (useful).	Is it possible to improve ground detection? Can more information be pulled from the images, HDR conversion?	Methods exist to convert to HDR from LDR, try those out to capture luminance data. Improve upon ground detection, and look into implementing Hosek-Wilkie sky model.
29th November	Ground detection improved using bias towards known correct ground position information. HDR conversion working and implemented to pipeline.	Can the Hosek-Wilkie model be implemented into the pipeline? Is HDR upscaling going to be a reliable luminesce source?	Continue looking into Hosek-Wilkie implementation. Write draft research report.
9th December	Hosek-Wilkie implemented using PBRT ImgTool. Research report draft written, final report just starting.	If time, can HDR upscaling be used to produce a reliable luminesce data resource? Can the pipeline be simplified?	Finish research report. Look into HDR upscaling as results from HDR conversion pipeline are very low resolution. Pipeline is coming along nicely.