Understanding the task

# Introduction

In Queens University Belfast I am enrolled upon a module called “Computer Science Challenges”. The task set is generally very difficult and is more to see just how well we get on with it rather than to complete the task. The main goal of doing this is less to complete the task but more in creating something of value.

This Blog post will hopefully give you something of value in the form of allowing you to follow on from the work I have done.

## The Task

People often make consistent superficial judgements of a strangers’ personality and life from their appearance. While these judgements can often be very inaccurate, the fact that many people can feel a similar way is interesting and can be used to reveal cultural bias.

The goal of this project is to create a realistic ‘character sheet’ as a json datastructure that defines a person and their appearance in an image.

By getting crowd workers to label fictional and real people we can analyse how people make judgements about others and how accurate and consistent such judgements can be.

You should try to judge an image of someone with the potential of producing measurements such as:

* Add career and education estimates (what job they might do, what subjects they studied at university) (net worth)
* Add romantic relationship estimates (given other images and descriptions identify the most likely partner) (potentially historical estimates e.g. many partners one, none etc.)
* Friendship relationships
* Family relationships
* Hobby and entertainment interests – what films, music, part time activities most interested in
* Medical conditions
* Religious views
* Weight, Height, clothing measurements
* Possible names the person might have (implying nationality, class, age etc.)

You could also attempt to train a stylegan-ada system to take an image and estimate properties of the person, also can adjust character sheet information to adjust the appearance

## Initial thoughts

My initial take on this was that it seems like an incredible idea, but a big undertaking, with a steep learning curve. I was particularly interested in the idea of using StyleGAN2-ada to take an image and estimate these properties and so this is what I focused on.

I hope that this blog post will help you understand the learning process that I went through.

## Table of contents

[Understanding the task 1](#_Toc65851142)

[Introduction 1](#_Toc65851143)

[The Task 1](#_Toc65851144)

[Initial thoughts 2](#_Toc65851145)

[Table of contents 3](#_Toc65851146)

[Frequently used terms 4](#_Toc65851147)

[StyleGAN2 6](#_Toc65851148)

[Overview 6](#_Toc65851149)

[Capabilities 6](#_Toc65851150)

[Relevance to the task 6](#_Toc65851151)

[StyleGAN2-ada 7](#_Toc65851152)

[Google Colaboratory 8](#_Toc65851153)

[Overview 8](#_Toc65851154)

[How I used it 8](#_Toc65851155)

[Advantages 8](#_Toc65851156)

[Limitations 8](#_Toc65851157)

[TfRecords 9](#_Toc65851158)

[Transfer learning 10](#_Toc65851159)

[Advantages of TL 10](#_Toc65851160)

[My Progress 11](#_Toc65851161)

[Understanding how to use Colab + StyleGAN2-ada together 11](#_Toc65851162)

[Long term goal 12](#_Toc65851163)

[What could be achieved 12](#_Toc65851164)

[Integration with peer’s work 12](#_Toc65851165)

[Other potential usages 13](#_Toc65851166)

[Useful Links for continuing project 13](#_Toc65851167)

# Frequently used terms

The field of data science can be daunting, I know it certainly was for me when I first started so I have compiled a list of commonly used terms and their meanings, you can also click on the word for a link to expand further.

* [Machine Learning](https://en.wikipedia.org/wiki/Machine_learning) – The study of computer algorithms that improve automatically from experience, they build a model based on sample data in order to make decisions / predictions without being explicitly programmed to do so
* [Neural Network](https://en.wikipedia.org/wiki/Neural_network)(AI) – A way of processing information patterns inspired by the way biological neural systems process data. They can be used to model complex relationships between inputs and outputs or to find patterns in data.
* [Deep Learning](https://en.wikipedia.org/wiki/Deep_learning) – Part of a broader family of machine learning methods. Deep refers to there being multiple layers used in the network, these layers are used to extract high level features from the input.
* [Latent Space](https://stats.stackexchange.com/questions/442352/what-is-a-latent-space) - abstract multi-dimensional space containing feature values that we cannot interpret directly, but it encodes an internal representation of externally observed events. I see it as basically how the network understands the data it has been given.
* [Vector](https://www.quora.com/What-is-a-vector-in-machine-learning?share=1) – An array of numbers describing a specific combination of properties. They are used to represent a point in the latent space.
* [Generative Algorithm](https://en.wikipedia.org/wiki/Generative_model) – A function that takes a vector in a latent space and produces a useful result like an image
* GAN encoder – Finds the vector in the latent space that would produce the most similar image to the input

# StyleGAN2

## Overview

[StyleGAN2](https://en.wikipedia.org/wiki/StyleGAN) itself is a generative adversarial network (GAN) which was brought forward from Nvidia researchers in December 2018.

A GAN is a class of machine learning frameworks in which two neural networks contest with each other in a game. Given a training set, this technique generates new data with the same statistics as the training set.

A video which helped me understand GAN’s can be found [here](https://www.youtube.com/watch?v=Sw9r8CL98N0)

## Capabilities

A GAN could therefore be used to generate new images that look similar to the ones in the dataset provided.

Researchers at Nvidia have used StyleGAN to produce images of people who don’t actually exist, by using a dataset of real people to train the GAN, this can be viewed on this quite eerie site: <https://thispersondoesnotexist.com/>, simply refresh the page to see a new person who does not exist.

## Relevance to the task

Labels can be attached to certain aspects of the image generation so that the user can help the GAN ‘understand’ real world characteristics.

For example, if you provided a dataset of people of varying ages, with their ages as labels to the image, you could then generate images of people defining what age you want them to look. See Below as to what the output could look like

This same system could be used to take an image and work out what the values of the labels would be. For example, in guessing the age of someone.

If we got this working, we could get the labels assigned to someone and from this formalise aspects of the image of a person and from this generate assumptions about them. Therefore, enabling us to create the “Character Sheet” of a person.

## StyleGAN2-ada

You may have noticed that I’m talking about StyleGAN2 when I’m working with StyleGAN2-ada.

StyleGan2-ada is simply an improved version and boasts many improvements over StyleGan2 these include:

* Significantly better results for datasets with less than 30k training images.
* Mixed precision support meaning ~1.6x faster training, ~1.3x faster inference, ~1.5x lower GPU memory consumption.
* A cleaner codebase which is therefore easier to work with.

The GitHub page for StyleGan2-ada can be found [here](https://github.com/NVlabs/stylegan2-ada)

# Google Colaboratory

## Overview

Basically, google Colaboratory is a free service which allows you to run your code on a virtual machine. Meaning you don’t need to worry about the power of your PC if you want to follow through on furthering this work.

## How I used it

I took the notebook found on [this GitHub page](https://github.com/woctezuma/stylegan2-ada) and edited it to suit my needs.

This allowed me to easily have each piece of code required to prepare my own dataset and train my network all on one page, and without using any of my computer’s resources (other than that required by the web page of course).

The link to my notebook can be found [here](https://colab.research.google.com/drive/13U2dZstwzfwE7vyiMVyNX76wYje8cfKD?usp=sharing)

## Advantages

* It Requires zero configuration and has many packages built in
* Gives free access to GPU’s for training networks
* You can easily share your work or pick up on the work of others

## Limitations

* Likes to shut off on its own once it has finished running (can be prevented)
* You have a maximum of 8 hours per session for the free version, so no very large datasets
* The data used is generally through your google drive folder which is limited to 15Gb without paying
* Generally the GPU is quite low power and therefore takes quite long to train
* You must have a google account

# TfRecords

TensorFlow Records or TfRecords are the file format in which StyleGAN2-ada reads datasets.

It is a binary file format which results in many performance benefits:

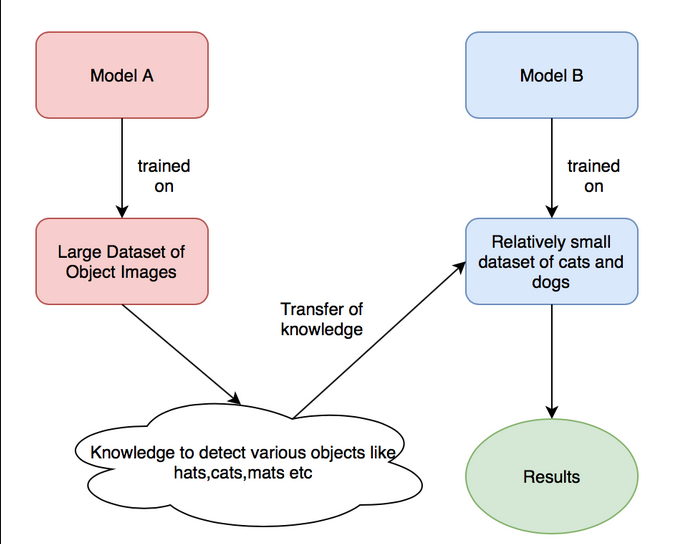
* Taking up less disk space
* Less time to copy
* Can be read much more efficiently from disk

It is also optimized for ease of use in working with tensorflow:

* It makes it easy to combine multiple datasets
* Splits larger datasets into batches, and only one batch needs to be loaded from the disk and processed at a time, saving resources

Info Source: [Medium Article](https://medium.com/mostly-ai/tensorflow-records-what-they-are-and-how-to-use-them-c46bc4bbb564)

# Transfer learning



Transfer Learning stores knowledge gained while solving one problem and applies it to a different but related problem.

In this specific example I have taken the ‘knowledge’ of the ffhq512 pretrained model, which is trained on human faces, and applied it to help generate images of dogs.

## Advantages of TL

* Saves on training time
* Better performance of neural networks (in most cases)
* Doesn’t need a lot of data

Info sources: [Wiki](https://en.wikipedia.org/wiki/Transfer_learning), [TL Article](https://builtin.com/data-science/transfer-learning)

# My Progress

I am currently at a stage where I have given a taken a pretrained network (one which has already been trained on a different dataset) and given it 3000 example images of dogs and from this produced its own crude ‘Dogs’



To see more information on how I got to this point please check out my how to guide.

I took on this task of training a StyleGAN2-ada network without labels so that later on I can train it with labels and eventually use this to produce my own images with my own parameters

## Understanding how to use Colab + StyleGAN2-ada together

The best way to learn how to do this yourself and further my work is to see an example. That is why I have fully annotated my colab notebook to help you understand and I highly recommend you check it out [here](https://colab.research.google.com/drive/13U2dZstwzfwE7vyiMVyNX76wYje8cfKD?usp=sharing).

Long term goal

# What could be achieved

The current aim of someone continuing my work (in my opinion) should be to get the TensorFlow labelling system integrated with my work and from this train a network to be able to recognise characteristics.

As a proof of concept, they should start by trying to take an image of a person and have the network try to guess what age they are. This could be achieved by firstly training the network on a dataset of people’s faces, with age labels attached to each image.

Once this has been achieved, we could then work towards the initial goal of producing a large amount of data, based on assumptions from an image and displaying it on the ‘real world character sheet’.

## Integration with peer’s work

Another Student, Nathan O’Donnell, is working on the same project however his focus is more on the character sheet side of things. So far, he has created a basic JSON Editor web application which can be used to read and update JSON data. In the project, it will be used to create, read, update and delete character sheet data. It is still in development but in future, more complicated UI elements will be able to be used to edit complicated and large pieces of data, his project can be found [here](https://github.com/nathan0donnell/character-sheet)

Someone could then take this work, and do one of two things:

1. They could produce an image using the network according to the label data provided in the web application (Creating image from character sheet)
2. Extract label data from an image and upload this data to the web application (Creating character sheet from image)

## Other potential usages

Another peer on the course is working on a ‘life simulation’ program, my work could potentially be integrated with their work in order to create an image of someone based upon their simulated characteristics.

## Useful Links for continuing project

* [Section in StyleGAN2-ada code for embedding labels](https://github.com/NVlabs/stylegan2-ada/blob/8c3fd8bac3e5a54a20e860fc163d6e67cc03c623/training/networks.py#L514)
* [Link to Nathan’s work](https://github.com/nathan0donnell/character-sheet)
* [StyleGAN2-ada GitHub issue on adding conditional labels](https://github.com/NVlabs/stylegan2-ada/issues/22)
* [Potentially using subfolders as labels when converting to tf records](https://androidkt.com/convert-tfrecords/)

Final note: if the link to my colab notebook in this document ever stops working then please access it via the GitHub repository.