Generating vector-based designs using GANs

Master’s Degree
Proposal
Computer Science

Writer: Raz Saroya
Advisor: Adi Shraibman
School of Computer Science - The Academic college of Tel Aviv-Yaffo, Israel
November 2020

Abstract

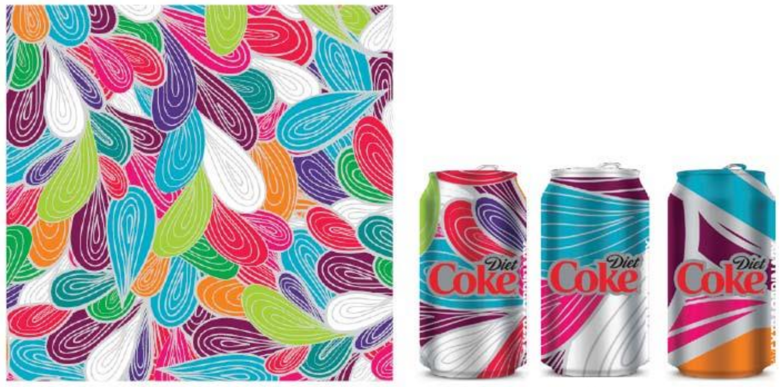
A **generative adversarial network (GAN)** is a relatively new class of machine learning framework. The idea is to learn to generate new data with the same statistics as a given training set. This opened a world of new possibilities.

For example, a GAN trained on images of human faces can generate new (artificial) images that look at least superficially authentic to human observers, having many realistic characteristics.

Objective

In recent years we have seen many applications of GANS in various areas, e.g. generating new examples for image datasets, generating cartoon characters.
The aim of this project is to use GANS to create new complex and high detailed vector-based designs, to be used as seeds for HP Mosaic.

HP Mosaic is an automatic creation of designs based on core patterns.
The core pattens are complex high detailed vector-based images, Mosaic algorithm randomly takes parts of the seed image on each iteration with random zoom and spin.
For example –



Seed file 200% 400% 800%

The purpose of this project is to help designers create unique and complex designs fitted to mosaic design requirements with the help of machine learning. The goal is to create complex vector patterns in SVG or PDF format based on trained data and user input.

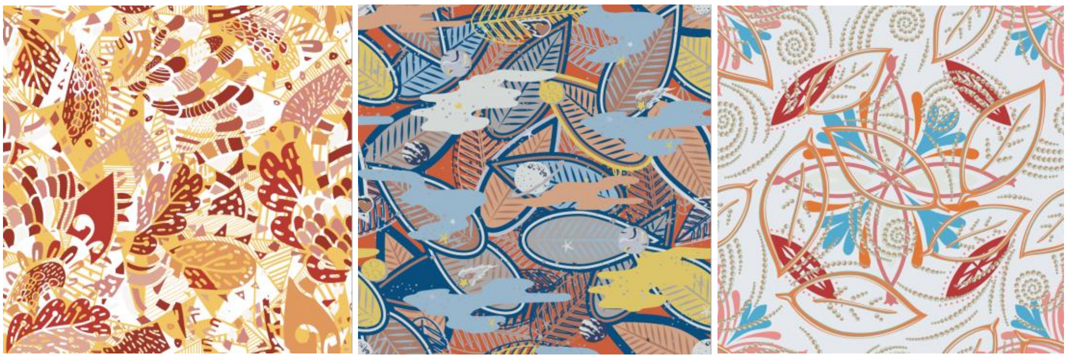
Today, designers have to look for the seed files from various stock photo libraries that fits mosaic design requirement – **complex graphics + in vector format + have the images that the designers like**. (need to fit 3 criteria)

Challenge today:

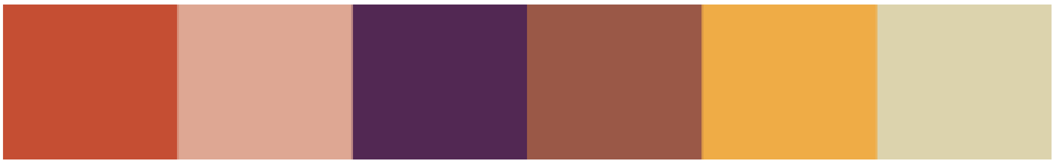
1) **Time consuming** - takes a lot of time to find the right pattern (3 criteria mentioned)
2) **Non-exclusive –** the same pattern or photo in stock photo library is used by many users. Brand designers usually worry that **someone else is using it!**
3) **Weak design outcome** - inexperience designers may **not** know how to choose the 'right' seeds that fit the Mosaic requirements and result into weak mosaic designs
4) **Costly to be unique**: in order to have unique patterns, designers must create the pattern from scratch / edit from stock graphics (take time & high cost)

The project’s challenge is to generate vector images using machine learning, which is basically like generating new code based on existing one. The benefit of vector graphic over raster graphic is that it’s not affected by scaling. Unlike raster graphic that constructed by limited number of pixels, which each pixel represents RGB color, vector graphic defined in terms of points on a Cartesian plane, which are connected by lines and curves to form polygons and other shapes.

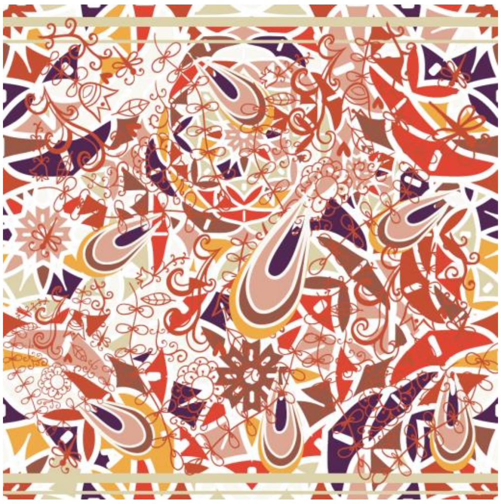
Example of usage - given the following vector images as train data -



and user color plate – (#c54e33, #dea793, #522853, #9a5847, #efac46, #dcd3ad)



The desire is to output image that combines patterns from all the input images and filled with the user input color plate –



Solution description

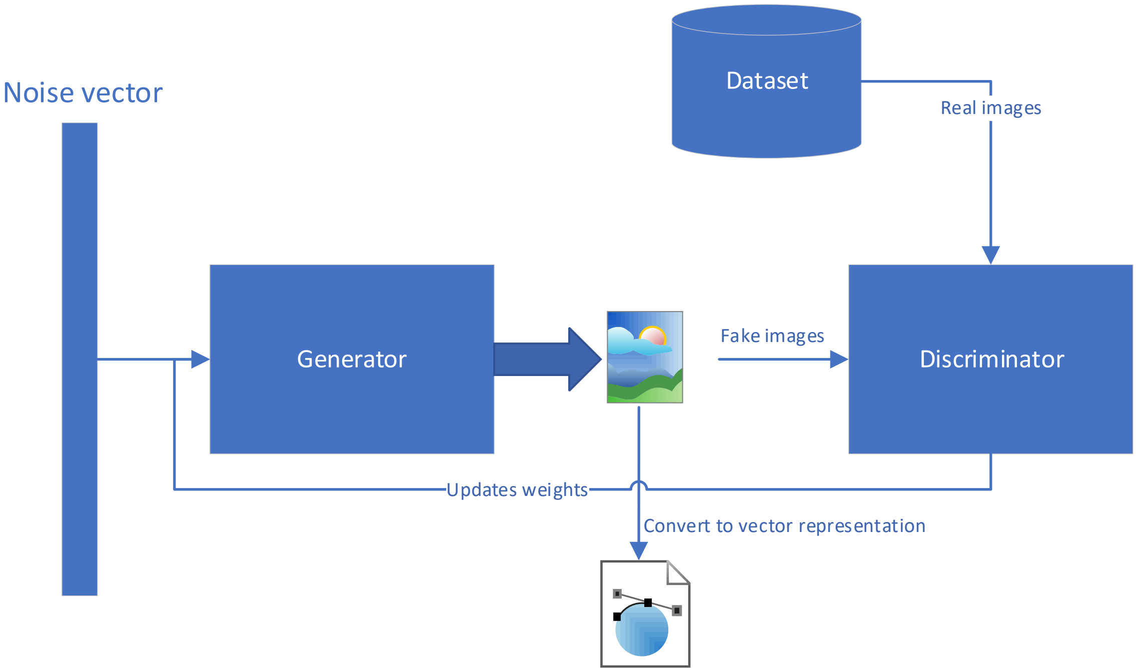
This project will include 2 stages:

Stage 1

The first stage of the project is to collect and create dataset of complex patterns images as the desired generated output image.

Then use common methods of GAN to train the dataset and compose raster images with the generator and convert them to vector-based images.

The is no good open source library to convert pixels to vectors so this step will require some efforts.

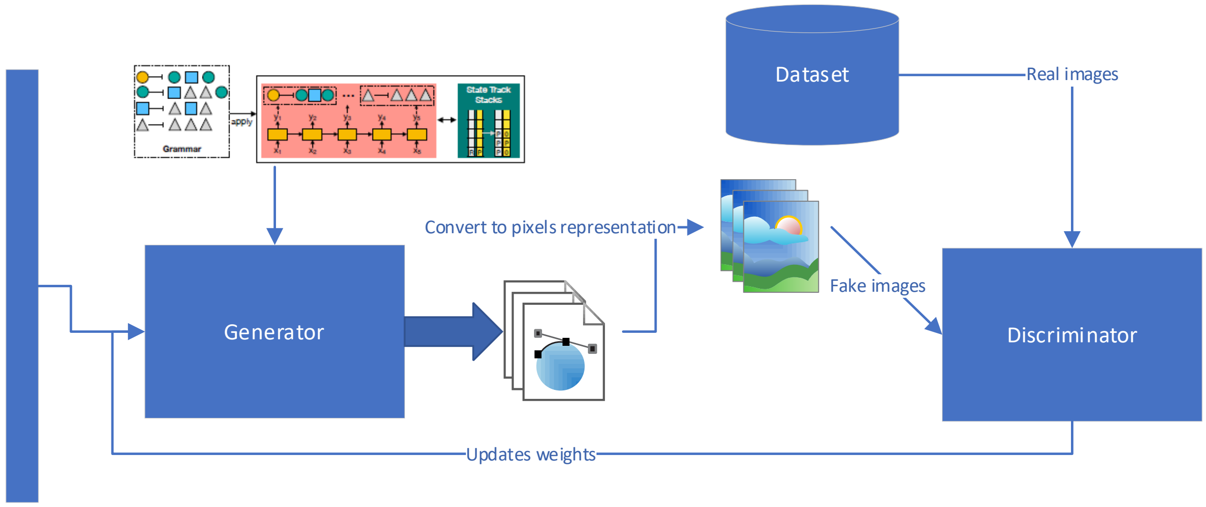


Stage 2

The second stage of the project is trying to use the generator to compose vector images in SVG format using TreeGAN, which is syntax aware sequence generation. This stage is challenging since the SVG format is very rich and complex compare to simple SQL queries which the syntax aware generation article is based on.

The process of using TreeGAN is not trivial and require creating syntax tree for the SVG format.

The architecture of the network:



Innovation

Most of the research conducted in this field based on image generation based on pixel matrixes. Stage 2 of this project is about generating code that describes the structure of an image which require sequence generation that must consider syntax and hard structure.

Success criteria

To define this project as successful at the end of the process, I would like to output vector format images that combines complex patterns as the input images.

Timeline

Stage	Due Date
Dataset preparation	01/02/2021 – 01/03/2021 (1 month)
Implement DCGAN and modify it for our task requirements	01/03/2021-15/03/2021 (2 weeks)
Experiments and tweaks with the DCGAN until it's outputs good results	15/03/2021-15/04/2021 (1 month)
Convert pixels images to SVG format	15/04/2021-06/05/2021 (3 weeks)

Create SVG format syntax tree for TreeGAN	06/05/2021-15/05/2021 (1 month)
Implement TreeGAN with SVG Format Syntax tree	06/06/2021-20/07/2021 (1.5 months)
Experiments and tweaks with the TreeGAN until it’s outputs good results	20/07/2021-20/08/2021 (1 month)
Submission	20/08/2021