Picture change with Evolutionary Algorithm

Evgenia Kivotova, BS17-08

March 2019

1 General Idea Description

This section describes the Evolutionary Algorithm used for image processing.

1.1 The notion of beauty

Although is true that each person has his own understanding of beauty, in this work, it is important to determine exactly what makes the images attractive to the human eye, first of all, since it is this thing that determined the final criterion for the algorithm. To do this, I used the ideas behind the science of art.

There are certain rules in the theory of the arts used when teaching academic drawing. In addition to concepts such as composition and technique, a lot of influence on the perception of the picture produces a color combination. But when composition and technique are hard to formulate with the language of math, the color combination became the perfect candidate to use.

1.2 Evolutionary Algorithm Description

1.2.1 Fitness Function

The fitness function is used to evaluate the score of picture beauty and force the algorithm to move towards the desired ideal. But which definition of an 'ideal' may be used in case of picture generation? In this work, the method of moving to some color palette was used.

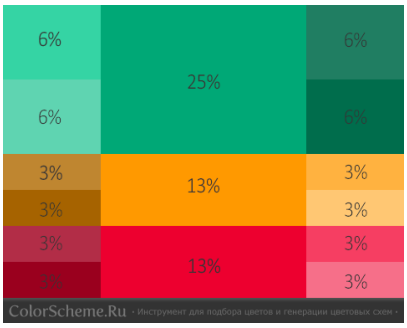


Figure 1.1: Triadic palette

As it is shown in Figure 1.1, each color combinations may be presented as a list of nodes, containing color code in RGB format with the desired percentage of it in a picture. Using this information it is easy to calculate the 'ideal' number of pixels needed to have for each color. So then the beauty score is nothing but the difference of current colors distribution with this computed ideal.

However, while having ideal color distribution, the picture still may be a mess, presented in Figure 1.2. To avoid this, the parameter called **'dispersion'** was used. Its value represents the percentage of neighbor pixel pairs that have different color value and plays the role of the controller, making pictures have larger solid color areas.

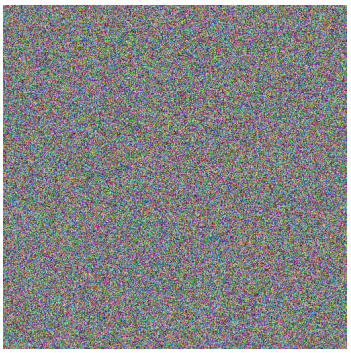


Figure 1.2: 'Ideal' noise

In general, if we define differense as **D**, dispersion value as **N**, total number of pixels as **P** and total number of colors used as **C**, the formula for beauty score **B** may be shown as:

$$B = \frac{PC}{1 + D + \frac{1}{2}PN}$$

1.2.2 Terminology

To make the next description of Evolutionary Algorithm more precise, let me introduce the next definitions of such algorithm terminology:

Chromosome - Full sized picture

Population - Set of pictures

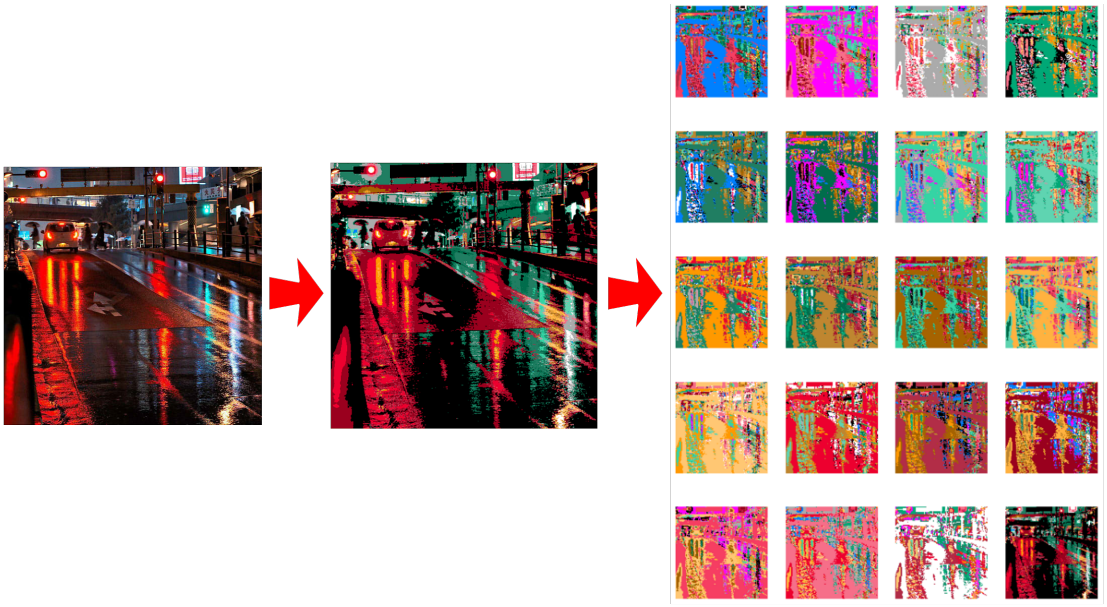
Now its time to go into algorithm details.

1.2.3 Initial Population

There were several ways to create the first population from the input image, such as picture division and duplication, but all these methods have their drawbacks: they either go against defined terminology or gives not good enough variety of choice at the beginning. Therefore, another approach was chosen.

Firstly, the input image was converted into a simplified version of itself, consisting only of colors that are defined in the desired palette. To make this converting better, the 'muted' additional colors may be added to the combination having wights of 0, so that they will add their value to fitness function result.

Secondly, the circle color switch performed, producing the copies of the initial image but with different color distribution. The set of resulting pictures then is called the first population.

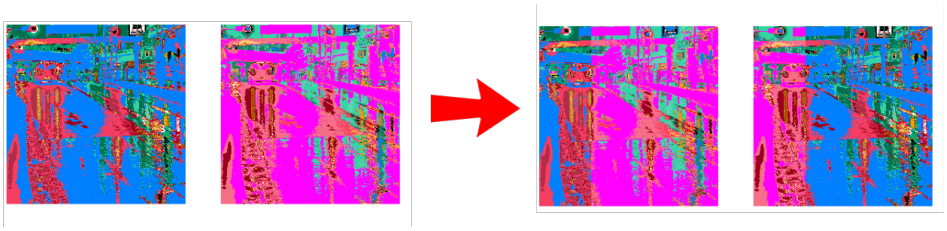


1.2.4 Selection

In the selection stage the 'wheel rotating' method was used based on calculated fitness value for each node.

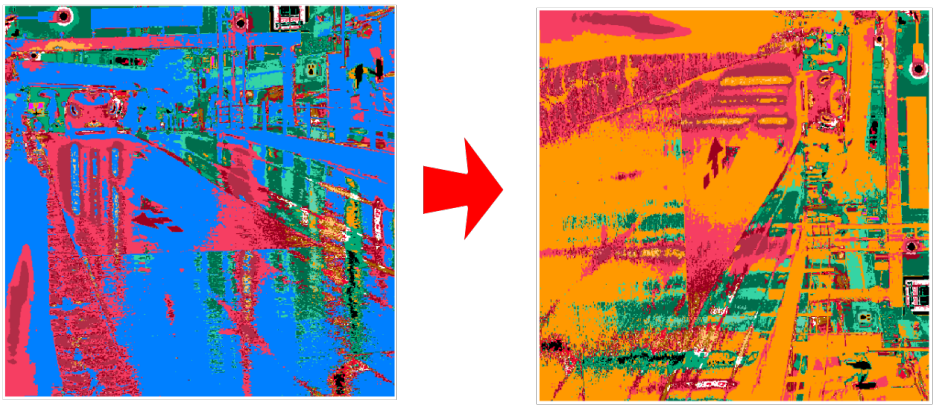
1.2.5 Cross Over

The cross over is rather straight forward: pairs of pictures are cut into 4 pieces, from which 2 new pictures produced, having one part of the first parent and other from second. Both parents and children are used as next population.



1.2.6 Mutation

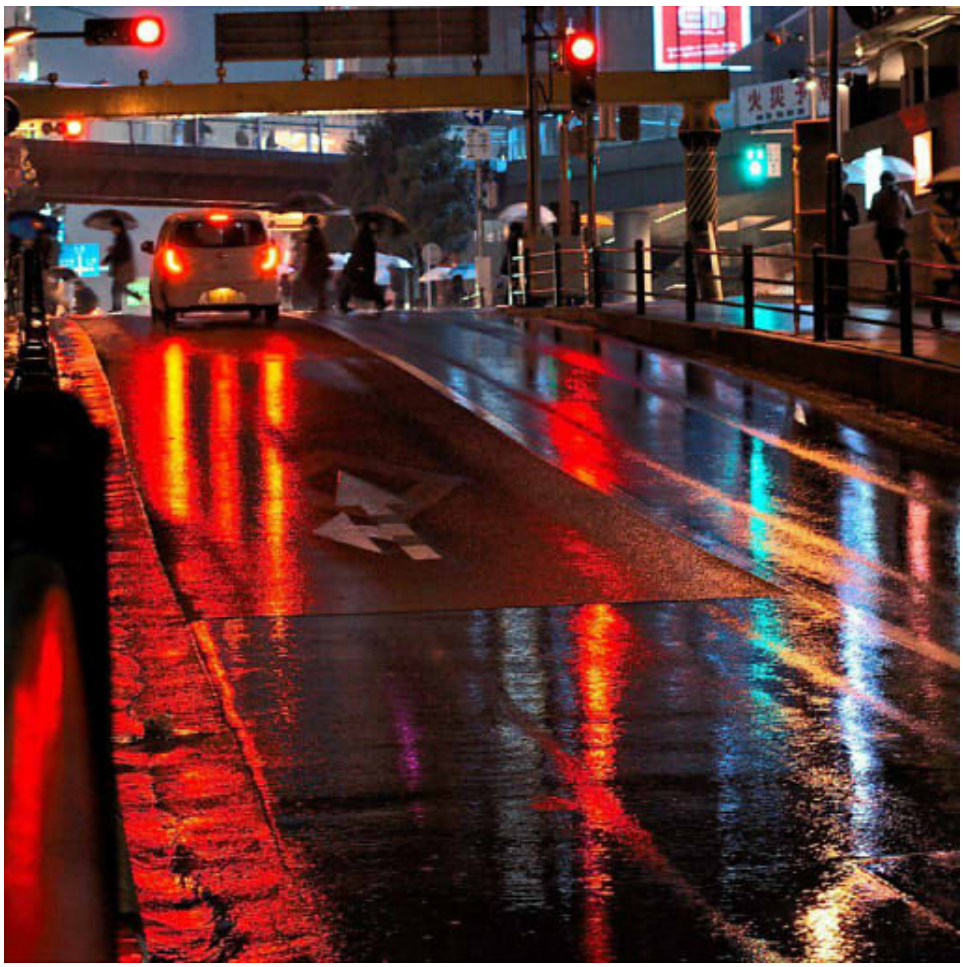
For the mutation stage, two random events were chosen: rotation and color change. Rotation provides the physical image reconstruction when color change - adds positive noise such as new color that did not initially present at the picture. Here is the example of mutation with rotation and blue to orange color change:



2 | Result Examples

It this section I provide the results of algorithm work with different initial settings. It was already said that the algorithm is expected to move towards a specified color combination, and here it may be exactly seen.

As initial image I took next 512x512 picture:



Then, I used colorscheme.ru site to peek pleasant to my eye color combinations. It was decided to have 200 populations for each run it total, as after this number of selections the resulting image already forms good enough color distribution. After algorithm run with different settings, I got the next results:

2.1 Example of 3 Triadic combinations:

The fundamental idea is that you take three colors which are somehow evenly spaced around the color wheel. I use the method of isosceles triangles.

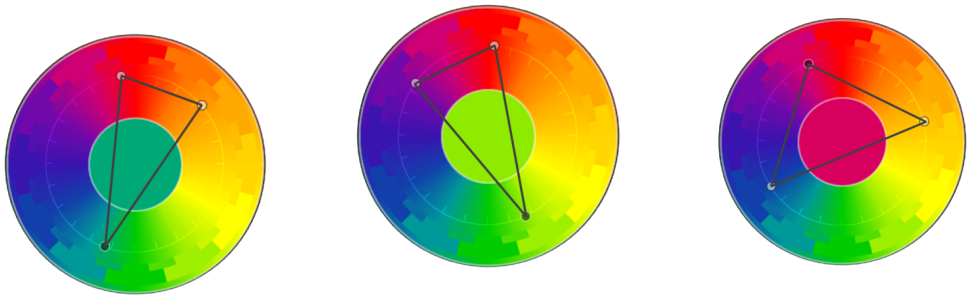


Figure 2.1: View of combinations on color wheel

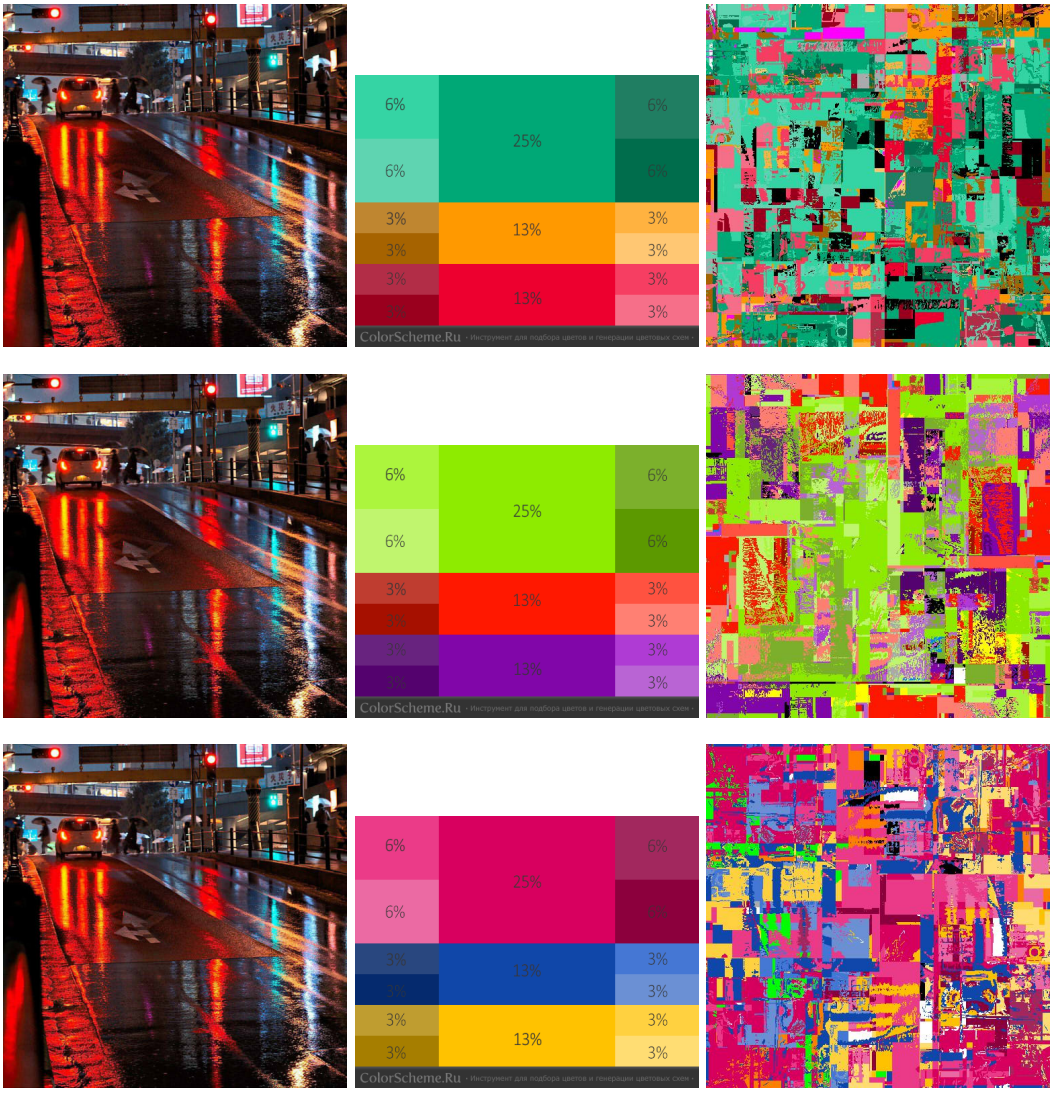
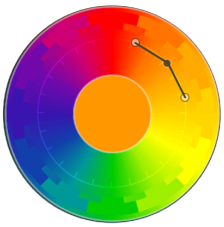


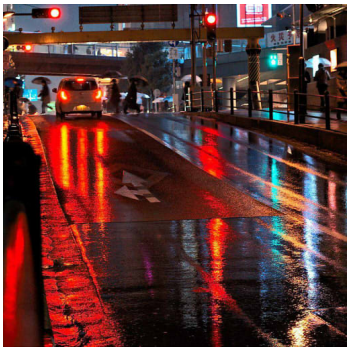
Figure 2.2: 3 Palettes with Result of Run

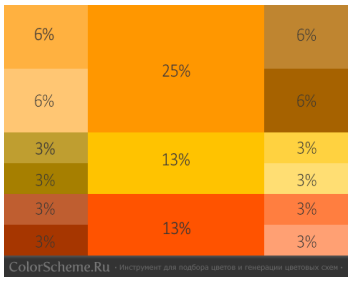
2.2 Example of Analogous Combination:

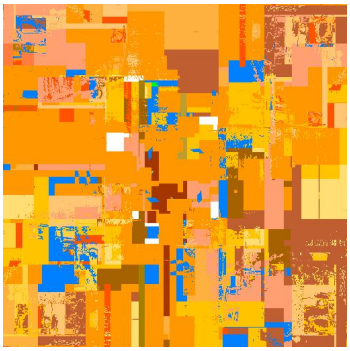
The analogous combination is represented by neighboring colors from the color wheel. This palette uses harmonizing colors, either in the warm or cool spectrum. For instance, red-orange-yellow combinations or green-blue-violet are more appealing than violet-green-orange.



Here I took Red-Orange-Yellow, warm combination as it has more clear color division then cold combinations, where blue, green and violet are hardly separated from each other.



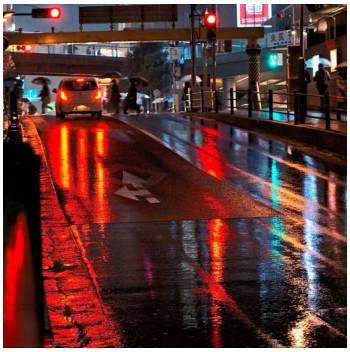


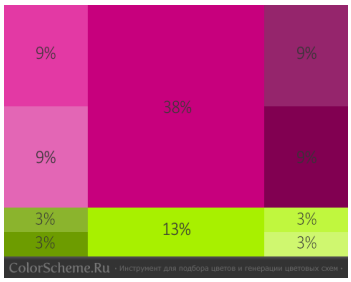


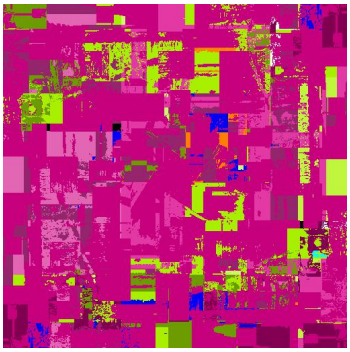
2.3 Example of Split Complementary Combination:

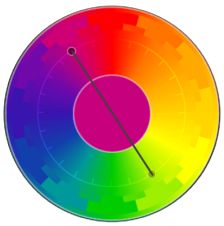
Split Complementary combination may be constructed by choosing one color and finding its complementary on the other side of the color wheel. Although the method is much simpler than in other combinations shown, the result looks much more appealing because of the bright contrast of these two colors.

Here it can be seen, how bright yellow-green color gives liveliness to the knotting set of different shades of purple.









3 | Real art simulation experiment

3.1 Piet Mondrian

There were many grate artists in the field of abstractionism: Mark Rothko, Jackson Pollock and many more. But here I want to focus the attention on artworks of Piet Mondrian, as it seems to me that my algorithm has the ability to get close enough to his stile.

How the works of Piet Mondrian may be described? His abstract creations has one thing in common : the color palette. Figure 3.1 explicitly shows what I am talking about.

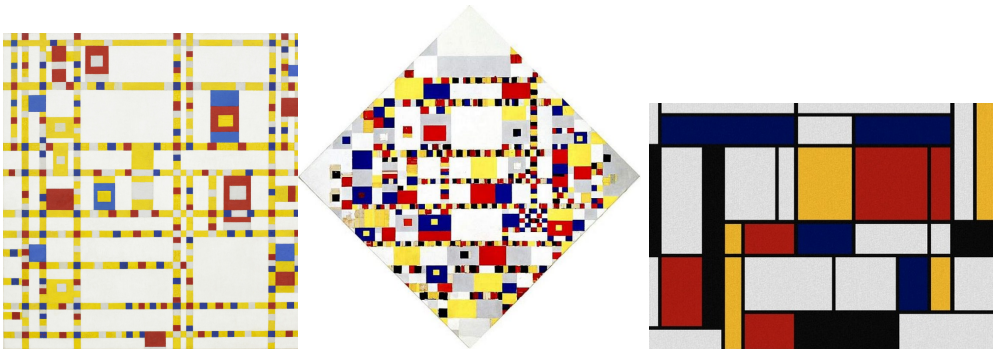


Figure 3.1: Piet Mondrian Artworks

After examples examining, the color palette was defined as follows:

30% - White #ffffff

25% - Red #d61010

20% - Yellow #fdd802

15% - Blue #0000a0

10% - Black #000000

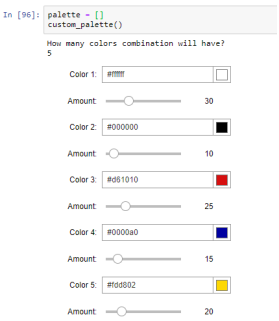
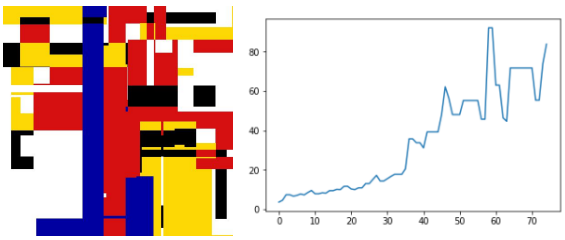


Figure 3.2: Custom settings

3.2 Experiment Result

75 populations based on purely white image gave the next result and fitness score growth. "The return of Boogie-Woogie".



4 References and Links

The color theory material:
https://en.wikipedia.org/wiki/Color_theory

The site where color palettes were chosen:
https://colorscheme.ru/

Color palettes used:
1. Triadic Combinations:
https://colorscheme.ru/export/screenshot/palette_3831Sw0w0w0w0.png.html
https://colorscheme.ru/export/screenshot/palette_2r31Sw0w0w0w0.png.html
https://colorscheme.ru/export/screenshot/palette_5h32Fw0w0w0w0.png.html

2. Analogous Combination:
https://colorscheme.ru/export/screenshot/palette_0L51Tw0w0w0w0.png.html

3. Split Complementary Combination:
https://colorscheme.ru/export/screenshot/palette_57227w0w0w0w0.png.html

Github with code and images:
https://github.com/Genvekt/EA-for-picture-change

Launched on Binder:
https://mybinder.org/v2/gh/Genvekt/EA-for-picture-change/master?filepath=Evolutionary%20Algorithm%20for%20picture%20generation.ipynb