UNIVERSITY OF TARTU

Peppa Pig and Voronoi Diagrams

Advanced Algorithmics project, fall 2017

Mikhail Papkov
Elizaveta Korotkova

Institute of Computer Science

Repository:
git.io/vNmnd

Contacts:
mikhail.papkov@gmail.com
elizaveta.korotkova@gmail.com

Abstract

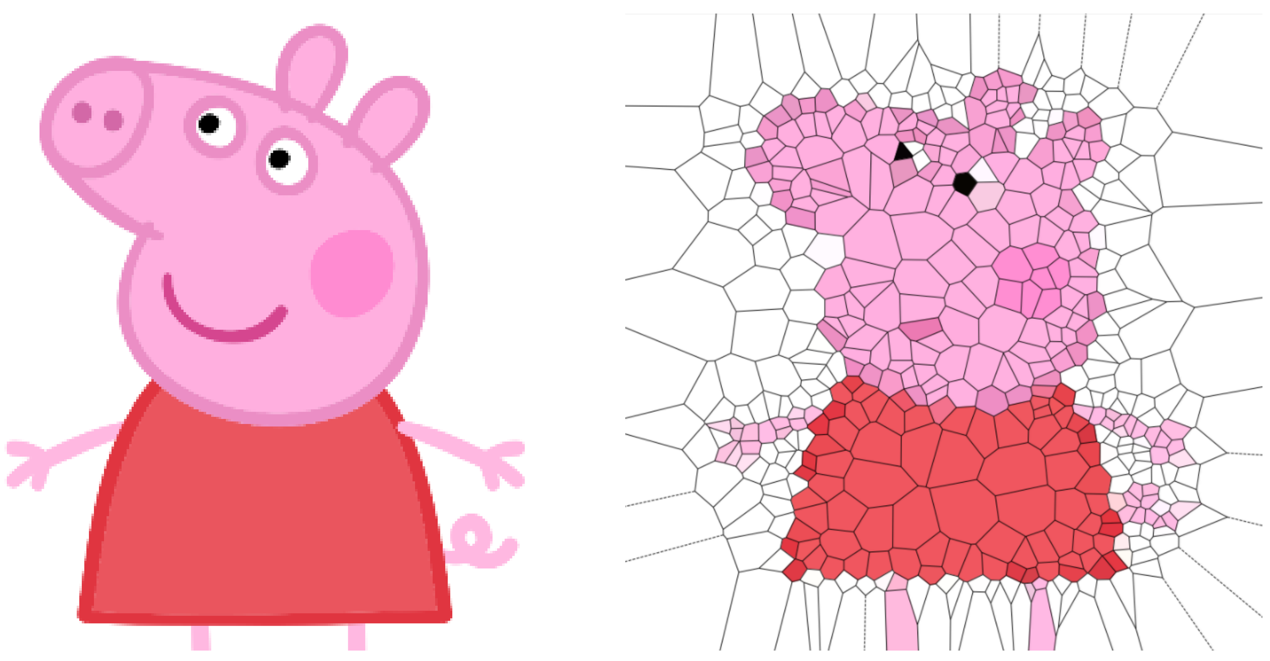
This poster presents the results of our participation in the project which was a part of the Advanced Algorithmics course at the Institute of Computer Science, University of Tartu. We have tried to approximate contour pictures with Voronoi diagrams using genetic optimization with different fitness functions.

Introduction

Goals

Our idea was to try to use Voronoi diagrams to approximate contour images and construct something similar to coloring pages. A genetic algorithm was used for the approximation.

There exists a solution, in which a Voronoi diagram of several hundred points is used to create a detailed polygon mesh of an image, using fuzzy k-means. We intended, however, to use relatively few points and to only approximate a contour of an uncolored image.



a: Source image **b:** Processed image

Figure 1: Fuzzy k-means algorithm for contour detalization (500 centroids)

Voronoi Diagrams

A Voronoi diagram is a partitioning of a plane into regions based on a set of predefined points $p_1 ... p_n$. The region of each point p_i in this set consists of every point in the Euclidean plane which is closer to p_i than to any other point of the set.

We used `scipy.spatial.Voronoi` to generate diagrams from points.

Image Preprocessing

The images were resized to 100×100 pixels for simpler shapes and to 150×150 for more complex ones. We used `skimage.morphology.skeletonize` to get thin, 1-pixel contours of the images.

Genetic Algorithm

A genetic algorithm is a metaheuristic used in optimization and search problems, inspired by the process of natural selection. Its principle relies on biological concepts like mutation, crossover and selection. Working process is the following:

- Create an initial population of Voronoi diagrams (individuals).
- Choose the top diagrams (individuals) according to some cost function.
- Crossover: create a new diagram (individual) using points (genes) of the best diagrams selected at the previous step.
- Introduce random mutations to some points (genes) of the new individual, creating a new generation.

Parameters

- Initialization (random; with centroids on the image skeleton)
- Number of centroids
- Population size
- Number of the best individuals chosen from each generation
- Mutation rate (how many centroids mutate)
- Mutation step (maximum by which the coordinates of a mutating centroid can change)
- Cost function (sum of distances vertex — skeleton; sum of top shortest distances vertex — skeleton; image overlap)
- Number of generations

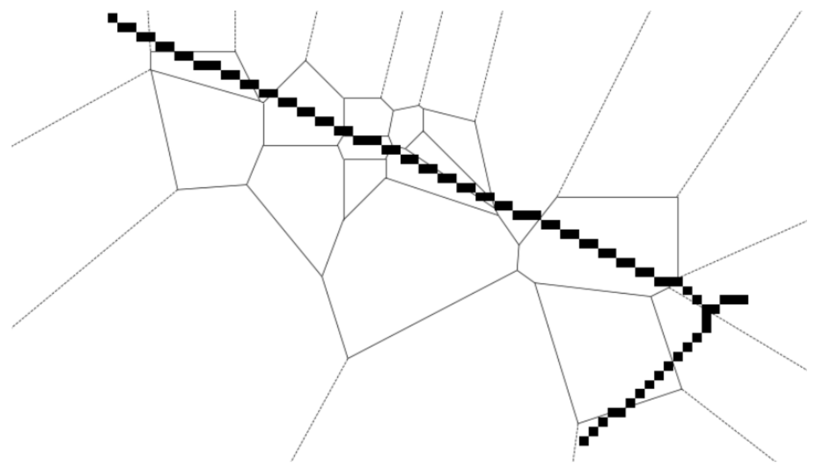


Figure 2: Algorithm demo for simple structure

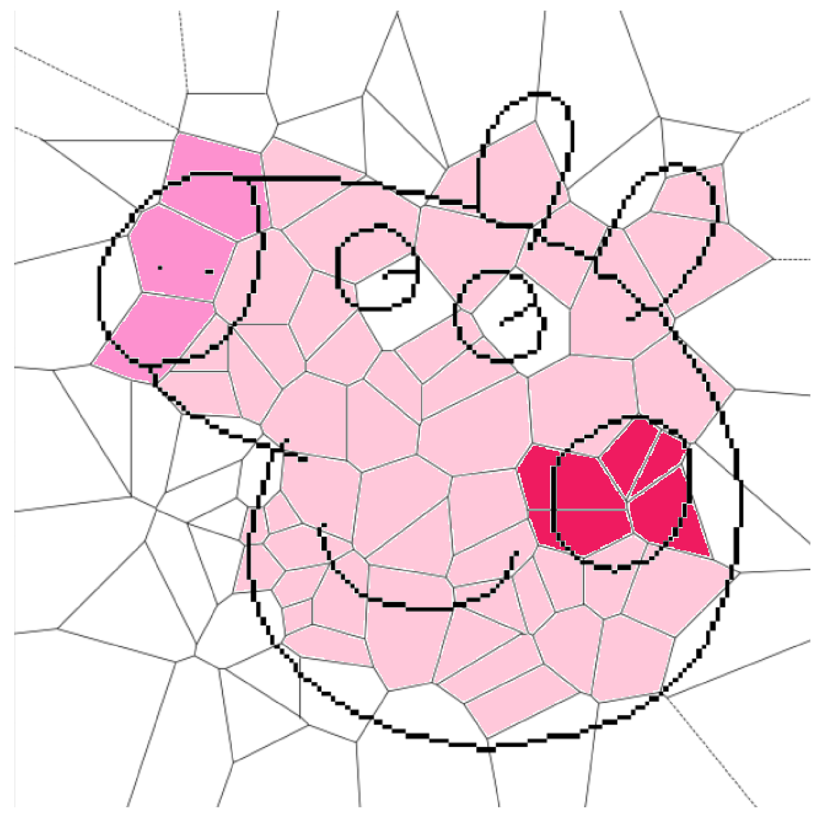
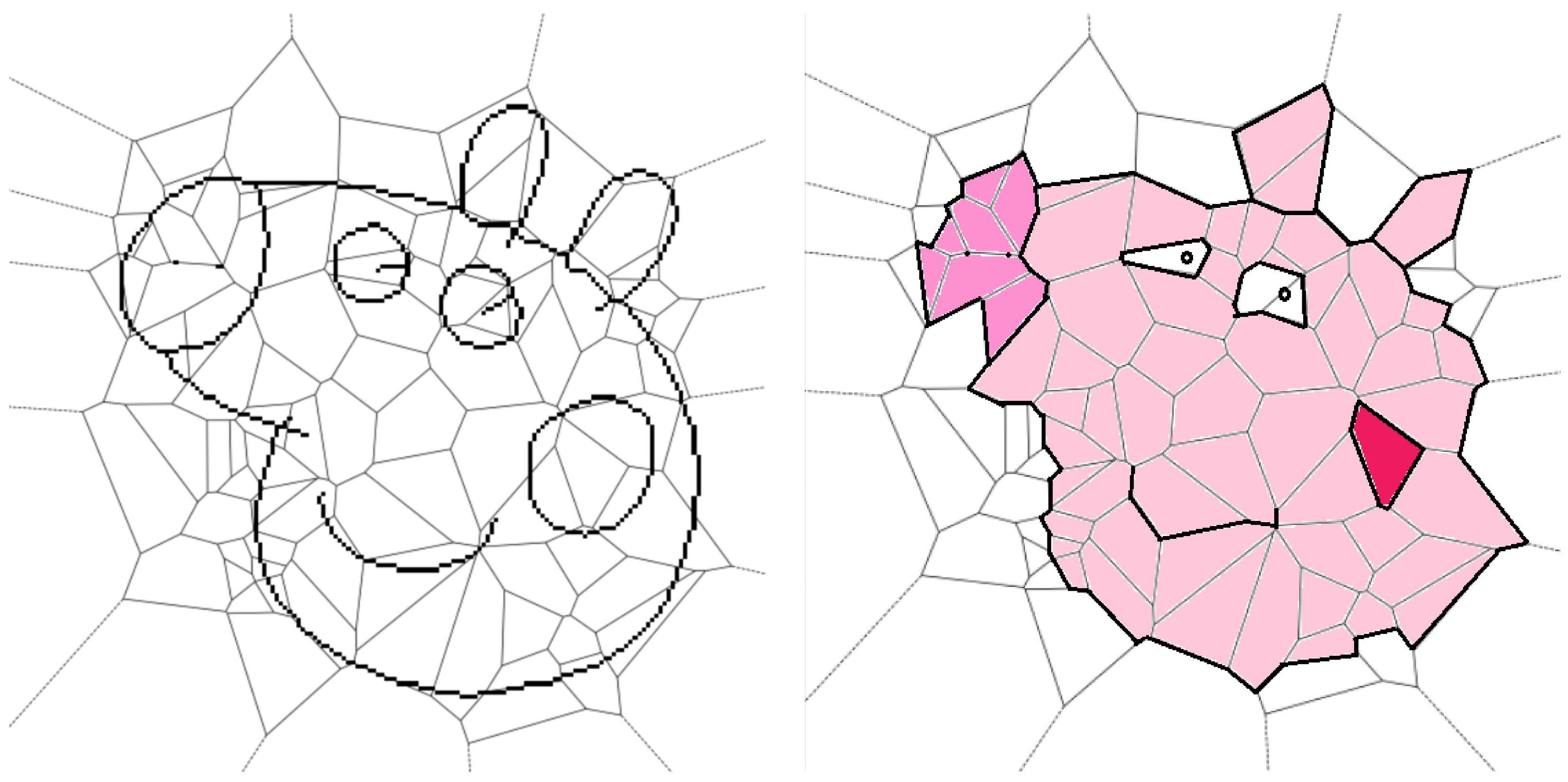


Figure 3: Initial diagram with random centroids

Results



a: Skeletonized contour on the diagram **b:** Colored diagram

Figure 4: Results of contour approximation (generations: 400, centroids: 100, population size: 30, mutation rate: 0.1, top selection: 5, max step: 3)

We used different sets of parameters and different cost functions to generate the diagrams. The algorithm does a more or less satisfactory job approximating simple shapes, while more complex and more finely detailed images are harder.

Acknowledgements

We would like to thank Archimedes Foundation for the Dora Plus scholarships granted.