



Instituto Tecnológico de Buenos Aires

Sistemas de Inteligencia Artificial

Trabajo Práctico 2

Algoritmos Genéticos

Ejercicio 1

Pensar (**no es necesario implementar**) cómo implementarían, mediante Algoritmos Genéticos, un programa que tome una imagen cuadrada y trate de representar de la mejor manera posible dicha imagen en un mapa de NxN caracteres ASCII, por ejemplo como se demuestra en la siguiente página: <http://www.nicassio.it/daniele/AsciiArtGenetic/>

Ejercicio 2

*Se busca implementar un compresor de imágenes un tanto peculiar. Deberemos implementar un motor de Algoritmos Genéticos que pueda recibir una imagen, y lograr la mejor aproximación a ella a través de **triángulos** sobre un canvas blanco.*

Nuestros únicos parámetros (no confundir con hiperparámetros) entonces serán la imagen a procesar y la cantidad de triángulos que queremos utilizar para aproximar esa imagen. Los triángulos deberán ser de un color uniforme pero pueden ser traslúcidos (RGBA, HSLA, ...).



Imagen a modo de ejemplificación input - output

Input

- Imágen
- Cantidad de triángulos
- Hiperparámetros de la implementación de Algoritmos Genéticos

Output

- Imágen generada
- Enumeración de triángulos (posición, color, ...)
- Métricas para análisis para defender su implementación (fitness, error, generaciones, etc...)

Implementar y resolver

- Implementar los métodos de selección vistos en clase
 - Elite
 - Ruleta
 - Universal
 - Boltzmann
 - Torneos (ambas versiones)
 - Ranking
- Implementar ambos criterios para crear nuevas generaciones
 - Tradicional
 - Sesgo Joven
- Decidir de qué manera(s) terminará la ejecución (máxima cantidad de generaciones, estructura, contenido, etc...)
- Justificar la estructura y la función de aptitud
- Decidir qué método(s) de cruce utilizarían en diferentes circunstancias y por qué.
Implementar al menos 2 métodos:
 - Cruce de un punto
 - Cruce de dos puntos
 - Cruce uniforme
 - Cruce anular
- Decidir qué método(s) de mutación utilizarían en diferentes circunstancias y por qué.
Implementar al menos 2 métodos:
 - Gen
 - MultiGen
 - Uniforme
 - No Uniforme

*(!) Es posible utilizar librerías externas para el manejo de imágenes, pero **no** para la implementación de Algoritmos Genéticos*

Entregable (digital)

- Código fuente
- Presentación
- Un archivo README explicando cómo ejecutar el programa

Preguntas que deben realizarse ANTES de experimentar...

- Cómo evaluo mi aproximación al dibujo?
- Qué es un individuo en este problema? Cuáles serían sus genes?
- Qué es el *fitness* en este problema?
- Cómo podría mutar un individuo?
- Cómo podría cruzar individuos para obtener descendencia? Esta cruza me genera descendientes con buenas probabilidades de obtener un mejor individuo?
- Cómo sería la versión más simple de esto?
- Qué tipo de imagen, y sobre todo cómo afecta la cantidad de triángulos a la performance si quiero evaluar rápidamente mi motor de AG?
- Alcanza implementar PARCIALMENTE los requerimientos de este trabajo práctico para evaluar mi motor de AG?

Opcionales

- La cantidad de triángulos parámetros es la cota máxima, y adicionalmente se recibe un error mínimo para considerar la imagen una buena réplica.
- Otros polígonos en vez de triángulos, u óvalos (x , y , r_x , r_y , θ).
- Otros métodos de cruza y mutación