

GENITIC ALGORITHM FOR OPTIMIZATION

```
import cv2

import numpy as np

import random

import matplotlib.pyplot as plt

# Genetic Algorithm Parameters

POP_SIZE = 20      # Population size

N_GENERATIONS = 30  # Number of generations

MUTATION_RATE = 0.2 # Probability of mutation

IMG_PATH = r"C:\Users\prami\OneDrive\Documents\Family.jpg"

def fitness(image, threshold):

    """Calculate fitness based on between-class variance (Otsu's method)."""

    _, binary = cv2.threshold(image, threshold, 255, cv2.THRESH_BINARY)

    hist = cv2.calcHist([image], [0], None, [256], [0,256]).ravel()

    total_pixels = image.size

    w0 = np.sum(hist[:threshold]) / total_pixels

    w1 = np.sum(hist[threshold:]) / total_pixels

    m0 = np.sum([i*hist[i] for i in range(threshold)]) / (np.sum(hist[:threshold])+1e-6)

    m1 = np.sum([i*hist[i] for i in range(threshold, 256)]) / (np.sum(hist[threshold:])+1e-6)
```

```
variance = w0 * w1 * (m0 - m1) ** 2  
return variance
```

```
def initialize_population():  
    return [random.randint(0, 255) for _ in range(POP_SIZE)]
```

```
def selection(population, scores):  
    """Roulette wheel selection."""  
    scores = np.array(scores)  
    probs = scores / (scores.sum() + 1e-6)  
    idx = np.random.choice(range(POP_SIZE), size=2, p=probs)  
    return population[idx[0]], population[idx[1]]
```

```
def crossover(p1, p2):  
    """Single point crossover."""  
    point = random.randint(0, 7)  
    mask = (1 << point) - 1  
    child1 = (p1 & mask) | (p2 & ~mask)  
    child2 = (p2 & mask) | (p1 & ~mask)  
    return child1, child2
```

```
def mutation(threshold):  
    """Random bit flip mutation."""  
    if random.random() < MUTATION_RATE:  
        bit = 1 << random.randint(0, 7)  
        threshold ^= bit  
    return max(0, min(255, threshold))
```

```

# =====
# Main GA Optimization
# =====

def genetic_thresholding(image):
    population = initialize_population()

    for generation in range(N_GENERATIONS):
        scores = [fitness(image, t) for t in population]

        new_population = []
        for _ in range(POP_SIZE // 2):
            p1, p2 = selection(population, scores)
            c1, c2 = crossover(p1, p2)
            new_population.append(mutation(c1))
            new_population.append(mutation(c2))

        population = new_population
        best_score = max(scores)
        best_threshold = population[np.argmax(scores)]

        print(f"Gen {generation+1}: Best Threshold = {best_threshold}, Fitness = {best_score:.2f}")

    return best_threshold

if __name__ == "__main__":
    # Load image in grayscale
    img = cv2.imread(IMG_PATH, cv2.IMREAD_GRAYSCALE)

    best_t = genetic_thresholding(img)

```

```
# Apply best threshold
_, final_img = cv2.threshold(img, best_t, 255, cv2.THRESH_BINARY)

# Show result
plt.subplot(1,2,1), plt.imshow(img, cmap="gray"), plt.title("Original")
plt.subplot(1,2,2), plt.imshow(final_img, cmap="gray"), plt.title(f'GA Threshold {best_t}')
plt.show()
```

output:

Gen 1: Best Threshold = 73, Fitness = 2550.92
Gen 2: Best Threshold = 6, Fitness = 2550.92
Gen 3: Best Threshold = 146, Fitness = 2543.74
Gen 4: Best Threshold = 203, Fitness = 2527.72
Gen 5: Best Threshold = 134, Fitness = 2543.74
Gen 6: Best Threshold = 70, Fitness = 2543.74
Gen 7: Best Threshold = 114, Fitness = 2549.20
Gen 8: Best Threshold = 66, Fitness = 2552.17
Gen 9: Best Threshold = 82, Fitness = 2552.17
Gen 10: Best Threshold = 82, Fitness = 2543.74
Gen 11: Best Threshold = 144, Fitness = 2543.74
Gen 12: Best Threshold = 178, Fitness = 2543.74
Gen 13: Best Threshold = 178, Fitness = 2408.85
Gen 14: Best Threshold = 210, Fitness = 2408.85
Gen 15: Best Threshold = 147, Fitness = 2537.01
Gen 16: Best Threshold = 146, Fitness = 2537.01
Gen 17: Best Threshold = 128, Fitness = 2537.01
Gen 18: Best Threshold = 148, Fitness = 2537.01
Gen 19: Best Threshold = 144, Fitness = 2537.01

Gen 20: Best Threshold = 184, Fitness = 2537.01
Gen 21: Best Threshold = 136, Fitness = 2537.01
Gen 22: Best Threshold = 144, Fitness = 2537.01
Gen 23: Best Threshold = 183, Fitness = 2537.01
Gen 24: Best Threshold = 140, Fitness = 2537.01
Gen 25: Best Threshold = 182, Fitness = 2537.01
Gen 26: Best Threshold = 170, Fitness = 2537.01
Gen 27: Best Threshold = 138, Fitness = 2532.83
Gen 28: Best Threshold = 136, Fitness = 2527.72
Gen 29: Best Threshold = 136, Fitness = 2488.61
Gen 30: Best Threshold = 154, Fitness = 2527.72