PSO for Object Recognition in Computer Vision

Documentation

Pesudocode

1. Initialize the parameters:
    a. Number of particles
    b. Maximum number of iterations
    c. Maximum number of unsuccessful iteration before explosion
    d. Maximum number of unsuccessful iteration before termination
    e. Parameters C1, C2, W
    f. X min and X max at each plane
    g. V min and V max at each direction
    h. Random initial values for:
        i. Each particle position – x, y, s (scale), r (rotation)
        ii. Each particle velocity in each direction
2. Evaluate the fitness value of each particle using fitness function
3. Set pbest (best solution) and gbest (best solution for swarm)
4. While a termination criterion is not met do:

$$Vel_i^{t+1} = w * Vel_i^t + c_1 * r_1 * (pbest_i^t - x_i^t) + c_2 * r_2 * (gbest_i^t - x_i^t)$$

$$x_i^{t+1} = x_i^t + Vel_i^{t+1}$$

```
Check if is inside constraints – amend if necessary
Calculate fitness value
If Xi > pbest:
    pbest = Xi
    If pbest > gbest:
        gbest = pbest
```

5. End when:
    a. Number of iterations > Max number of iterations
    b. Number of iterations without gbest update > Max number of iterations since update
6. Explode when gbest doesn't change for Maximum number of unsuccessful iteration before explosion – start PSO over but save gbest value

Fitness function

$$\boldsymbol{ERR_{max} = 2^{nbits} * \left((m * n) - Pinv\right)}$$

$$\boldsymbol{ERR_{CALC} = \sum_{i=0}^{n}\sum_{j=0}^{m} |RI(i,j) - LI(I,J)|}$$

$$\boldsymbol{ERROR = \frac{ERR_{max} - ERR_{calc}}{ERR_{max}}}$$

$$\boldsymbol{I = y + s * (ddX * sin(-r) + ddY * cos(r))}$$

$$\boldsymbol{J = x + s * (ddX * cos(-r) + ddY * sin(r))}$$

$$\boldsymbol{ddX = j - \frac{Width_{RI}}{2}}$$

$$\boldsymbol{ddY = i - \frac{Height_{RI}}{2}}$$

Python implementation