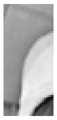
lenaGray = ;

lenaSubGray = ;

```
pic1 = lenaGray; pic2 = lenaSubGray;
```

```
(*参数设定*)
number = 20(*粒子群个数*); wmax = 0.9; wmin = 0.1; (*惯性系数*)T = 100
(*代数*); c1 = 2; c2 = 2; d = c1 + c2; {m, n} = ImageDimensions[lenaGray];
{x, y} = ImageDimensions[lenaSubGray];
amin = 1; amax = m - x + 1;(*参数范围*)
bmin = 1; bmax = n - y + 1;
vmax = Max[x, y]; vmin = -vmax;
a = Array[0, number];
b = a; aBest = a; bBest = a;
va = a; vb = a; p = a;
fitnessGray[u_, v_] := Total[Abs /@ (Flatten@ImageData@ImageDifference[u, v])]
```

```
For[i = 1, i <= number, i++, a[[i]] = Round[RandomReal[1] (amax - amin) + amin];
  aBest[[i]] = a[[i]]; b[[i]] = Round[RandomReal[1] (bmax - bmin) + bmin];
  bBest[[i]] = a[[i]]; va[[i]] = Round[(vmax - vmin) RandomReal[1] + vmin];
  vb[[i]] = Round[(vmax - vmin) RandomReal[1] + vmin];
  pic3 = Image@ImageData[pic1][[b[[i]] ;; (y - 1 + b[[i]]), a[[i]] ;; (x - 1 + a[[i]])]];
  p[[i]] = fitnessGray[pic2, pic3]];

{minp, index} = {Min[p], Position[p, Min[p]] // Flatten // First};
taBest = a[[index]];
tbBest = b[[index]];

pBest = Array[0, T]; ABest = pBest; BBest = pBest;
```

```
timeList = Table[For[t = 1, t ≤ T, t++, w = wmax - t * (wmax - wmin) / T;
      For[i = 1, i < number, i++, va[[i]] = w * va[[i]] + c1 * RandomReal[1] *
           (aBest[[i]] - a[[i]]) + c2 * RandomReal[1] * (taBest - a[[i]]); (*更新速度*)
        a[[i]] = a[[i]] + va[[i]]; (*更新位置*)a[[i]] = Round[a[[i]]];
        If[a[[i]] > amax, a[[i]] = amax, If[a[[i]] < amin, a[[i]] = amin];
         vb[[i]] = w * vb[[i]] + c1 * RandomReal[1] * (bBest[[i]] - b[[i]]) +
           c2 * RandomReal[1] * (tbBest - b[[i]]); (*更新速度*)
         b[[i]] = b[[i]] + vb[[i]]; (*更新位置*)b[[i]] = Round[b[[i]]];
         If[b[[i]] > bmax, b[[i]] = bmax, If[b[[i]] < bmin, b[[i]] = bmin]]; pic3 =
          Image@ImageData[pic1][[b[[i]] ;; (y - 1 + b[[i]]), a[[i]] ;; (x - 1 + a[[i]])]];
         pNew = fitnessGray[pic2, pic3]; If[pNew < p[[i]], p[[i]] = pNew;
          aBest[[i]] = a[[i]]; bBest[[i]] = b[[i]],
          If[p[[i]] < minp, minp = p[[i]]; taBest = a[[i]]; tbBest = b[[i]]]]]
      ]; pBest[[t]] = minp; (*每代的全局最优解*)
      ABest[[t]] = taBest; BBest[[t]] = tbBest]; // Timing, {10}]
```

```
{{1.825, Null}, {1.934, Null}, {1.825, Null}, {1.748, Null}, {1.825, Null},
 {1.95, Null}, {1.794, Null}, {1.607, Null}, {1.887, Null}, {1.857, Null}}
```

```
meanTime10 = Mean@timeList[[All, 1]]
```

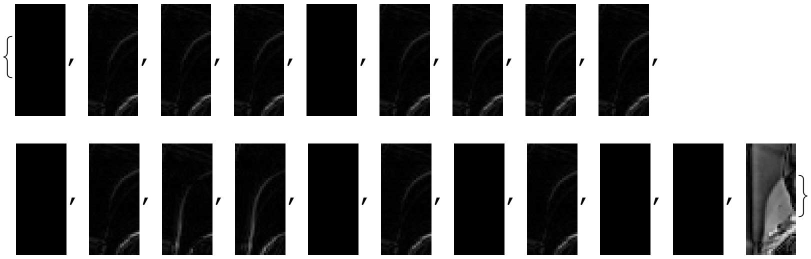
```
1.8252
```

```
Dynamic[{pic2, pNew(*匹配度*), pic3, taBest,
   tbBest, pointsToPlot = Table[pBest[[i]], {i, 1, T - 1}];
ListLinePlot[pointsToPlot, Filling → Axis]}]
```

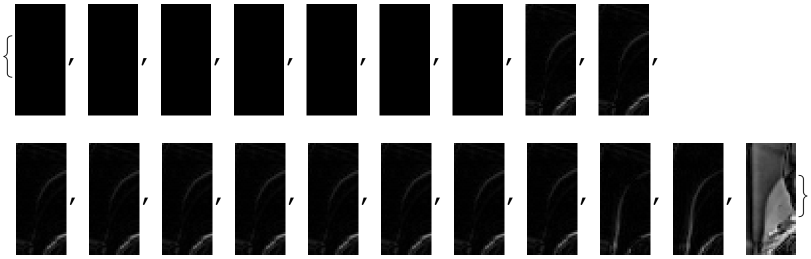
```
{pic2, pNew, pic3, taBest, tbBest,
 ListLinePlot[Table[pBest⟦i⟧, {i, 1, -1 + T}], Filling → Axis]}
```

```
picSubFound = SortBy[mostMatchedlist =
     Table[pic3 = Image@ImageData[pic1][[bBest[[i]] ;; (y - 1 + bBest[[i]]), aBest[[i]]
           ;; (x - 1 + aBest[[i]])]], {i, 20}], fitnessGray[pic2, #] &] // First
{"匹配差异值=", fitnessGray[pic2, picSubFound]} // TableForm
pos = Position[mostMatchedlist, picSubFound] // Flatten // First;
centerPosition = {aBest[[pos]], bBest[[pos]]} // Flatten;
```

```
ImageDifference[pic2, #] & /@ mostMatchedlist
```



```
ImageDifference[pic2, #] & /@ SortBy[mostMatchedlist =
   Table[pic3 = Image@ImageData[pic1][[bBest[[i]] ;; (y - 1 + bBest[[i]]),
        aBest[[i]] ;; (x - 1 + aBest[[i]])]], {i, 20}], fitnessGray[pic2, #] &]
```



```
imMatched =
  ImagePad[picSubFound, {{centerPosition[[1]], ImageDimensions[pic1][[1]] -
       centerPosition[[1]] - ImageDimensions[picSubFound][[1]]},
     {ImageDimensions[pic1][[2]] - centerPosition[[2]] -
       ImageDimensions[picSubFound][[2]], centerPosition[[2]]}}, Brown];
```

```
ImagePad[ImageAssemble@Partition[{ImageMultiply[imMatched, pic1],
     imMatched, pic1, ImageDifference[imMatched, pic1]}, 2], 20]
```