

Genetski algoritam za rešavanje puzzle

Nemanja Milićević RA 144/2013

Soft kompjuting 2016/17 Fakultet tehničkih nauka, Univerzitet u Novom Sadu

Abstrakt

Za ovaj projekat, implementiran je genetski algoritam koji rešava puzzle, opisan u "A Genetic-Based Solver for Very Large Jigsaw Puzzles" od Sholomon et al [1]. Ova metoda uključuje formiranje potomaka na osnovu roditelja, gdje se iz generacije u generaciju prosleđuju najbolje jedinke. Algoritam opisan u gore pomenutom radu može da riješi puzzle do 22,834 dijela u razumnom vremenu. Algoritam iz ovog projekta nije sposoban da riješi puzzle iste veličine kao u gore pomenutom radu za slično vrijeme, ali je sposoban da riješi manje puzzle sa visokom preciznošću.

Uvod

Cilj ovog projekta je da se reimplementira algoritam opisan u [1]. Navedeni rad opisuje metodu koja oponaša prirodnu selekciju tako što odabere puzzle, tj. Sliku podijeljenu male kvadratne dijelove, i kroz mnogo iteracija opisanog algoritma, rekonstruiše originalnu sliku. Prvo, populacija početnih rešenja, nasumično izgenerisanih slika na osnovu ulazne slike je izgenerisana. Zatim, određeni broj generacija se određuje mjera prilagođenosti jedinki u populaciji *fitness* funkcijom, i nova populacija se formira sa potomcima koji su rezultat ukrštanja odabranih roditelja iz prethodne generacije. Jedinke sa većom mjerom prilagođenosti imaju veće šanse da budu odabrane u fazi selekcije za reprodukciju. Konačno rešenje je jedinka koja ima najveću vrijednost *fitness* funkcije u poslednjoj generaciji.

```
1. population ← generate 1000 random chromosomes
2. for generation_number = 1 → 100 do
3.   evaluate all chromosomes using the fitness function
4.   new_population ← NULL
5.   copy 4 best chromosomes to new_population
6.   while size(new_population) ≤ 1000 do
7.     parent1 ← select chromosome
8.     parent2 ← select chromosome
9.     child ← crossover(parent1, parent2)
10.    add child to new_population
11.   end while
12.   population ← new_population
13. end for
```

Genetski algoritam

Genetski algoritam (skraćeno GA) je pretraga optimalnog rešenja unutar prostora rešenja. Pošto je ispitivanje ispravnosti svih mogućih rešenja praktično nemoguće, GA nude heuristiku za optimizaciju inspirisanu teorijom prirodne selekcije.

Prvo se nasumično generiše početna populacija kandidata za rešenje, takozvanih *hromozoma*. Svaki hromozom je potencijalno rešenje za problem, predloženi raspored dijelova puzzle. Nakon toga, primjenjuju se razni biološki inspirisani operatori kao što je selekcija, reprodukcija ili mutacija. Upotreba ovih operatora postepeno poboljšava rešenja u populaciji i eventualno dovodi do optimalnog rešenja.

Da bi oponašao prirodnu selekciju, stopa reprodukcije hromozoma je direktno srazmjerna njegovoj mjeri prilagođenosti ili *fitness* - u. Mjera prilagođenosti se dobija *fitness* funkcijom i predstavlja kvalitet datog rešenja. Bolja rešenja imaju relativno više potomaka nego ostala i dobri hromozomi imaju veću šansu da formiraju još bolje rešenje u sledećoj generaciji. Operacija reprodukcije zvana *crossover* bi trebala da prosljedi dobre osobine sa roditelja na potomke. Uspjeh GA je uglavnom zavisn od izbora pogodne reprezentacije hromozoma, operacije ukrštanja i fitness funkcije. Fitness funkcija mora tačno otkriti hromozome koji sadrže obećavajuće dijelove rešenja i prosljediti ih u sledeću generaciju.

Selekcija

Za selekciju se koristi funkcija direktno srazmjerna vrijednosti fitness funkcije za dati hromozom, ili takozvana rulet selekcija. Vjerovatnoća da se izabere neka jedinka je

$$p_i = \frac{f_i}{\sum_{j=1}^N f_j}$$

Gdje je f_i fitness hromozoma.

Evaluacija hromozoma

Za evaluaciju pojedinačnog rešenja se koristi mjera različitosti opisana ispod. Pretpostavka je da susjedni dijelovi u puzzle imaju slične boje na ivicama spajanja i da dva dijela puzzle predstavljena matricom $K \times K \times 3$, gdje je K širina/visina dijela (u pikselima). Njihova mjera različitosti se može izraziti kao:

$$D(x_i.b, x_j.d) = \sqrt{\sum_{k=1}^K \sum_{ch=1}^3 (x_i(k, K, ch) - x_j(k, 1, ch))^2}$$

Gdje je su x_i i x_j susjedni dijelovi i x_j je desno od x_i .

Fitness funkcija

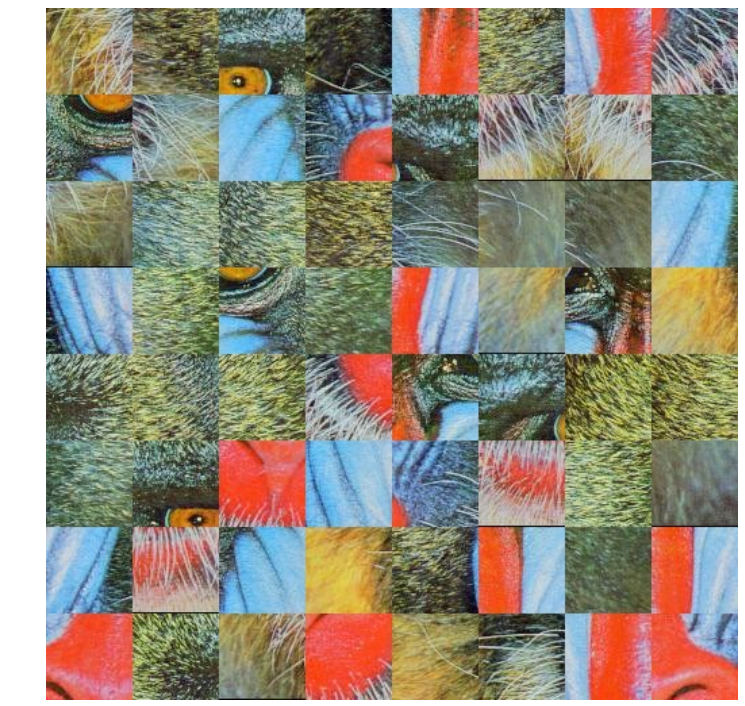
Fitness funkcija je definisana kao suma mjera različitosti između svaka dva susjedna dijela puzzle u hromozomu. Može se predstaviti sledećom formulom:

$$\sum_{i=1}^N \sum_{j=1}^{M-1} (D(x_{i,j}, x_{i,j+1})) + \sum_{i=1}^{N-1} \sum_{j=1}^M (D(x_{i,j}, x_{i+1,j}))$$

Rezultati

Algoritam je pokrenut više puta za svaku sliku iz skupa podataka sa 1000 jedinki u populaciji i sa 100 generacija. Iako je pokazao visoku tačnost, vrijeme izvršavanja je znatno veće nego u radu opisanom u [1]. Slike su izdijeljene u kvadratke dijelove dimenzija 28x28 piksela. Algoritam je testiran sa slikama sa sledećim brojem dijelova: 16, 32, 64, 128, 256, 512, 1024. Vrijeme izvršavanja je zavisno od broja jedinki u populaciji, broja generacija i veličini slike. Rezultati su pokazali sličnu tačnost kao i u [1]. Potrebno je naglasiti da nije bilo nikakvog proširenja za paralelizam i da je korišten Intel i7 6700HQ procesor.

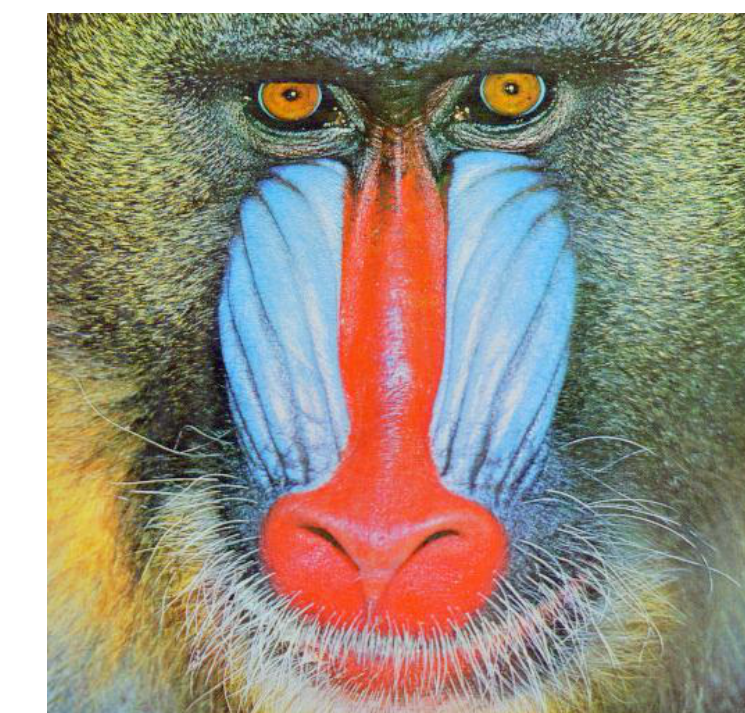
Sledeći primjer je slika od 64 dijela za koju je pokrenut algoritam sa 50 jedinki i 10 generacija.



Slika 1. Početna slika



Slika 2. Prva generacija



Slika 3. Poslednja generacija

Zaključak

Algoritam implementiran u ovom projektu se pokazao kao uspješan u rešavanju puzzle raznih dimenzija. Iako je preciznost visoka, vrijeme izvršavanja je znatno veće nego opisano u [1]. Ovaj algoritam se može koristiti za rešavanje puzzle različite težine i kao takav je doprinos skupu algoritama za rešavanje puzzle.

Literatura

- [1] D. Sholomon, O. David, N. Netanyahu. A Genetic Algorithm Based Solver for Very Large Jigsaw Puzzles. In 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1767-1774.