

Segmentacija tkiva korišćenjem MCL algoritma

Klasterovanje (clustering) je tehnika istraživanja podataka koja otkriva objekte (koji se opisuju određenim atributima) i deli ih u grupe (klasterne) čineći ih preglednijim i korisnijim.

Ljudsko oko lako može prepoznati prirodne klasterne, ali je ovo i više nego zahtevan zadatak za računar. Postoji dosta različitih algoritama za klasterovanje (k-means, GMM..) svaki sa razlicitom složenosti i razlicitom primenom u zavisnosti od prirode problema.

Problem koji ovde razmatramo je segmentacija respiratornog tkiva. Ulazni set podataka se sastoji od 800 slika tkiva plucne maramice. Zanima nas kako na zadatom setu podataka možemo odvojiti jezgro, citoplazmu i ostale celijske organele u klasterne, da bismo dalje posmatrali promene koje se odvijaju na ovim organelama. Ukoliko prevedemo ovaj problem u formu koja je čitljiva računaru, cilj nam je da obojimo svaku od slika tako da njene komponente budu obojene razlicitim bojama, npr. citoplazma zutom, jezgro i organele braon, a pozadina svetlo plavom bojom.

Svaku sliku možemo predstaviti kao funkciju (vektorski) sa određenim ponasanjem ili kao graf (rasterski). Konkretno, procesuiranje nad slikama predstavljenim kao grafovima funkcioniše na principu razlike u pikselima, odnosno svaki piksel (zadat npr. pomoću RGB skale kao vektor ili tačka u trodimenzionalnom prostoru) predstavljamo čvorom grafa, a kažemo da između dva čvora u grafu postoji ivica ako su oni na rastojanju manjem od unapred zadanog ρ . Ovo rastojanje se naziva Euklidsko rastojanje razlike.

Kod CCL (connected component labeling) algoritma, pikseli se grupišu u komponente na osnovu sličnosti u njihovim osobinama (boji). Svi pikseli u povezanoj komponenti dele sličan intenzitet i na neki način su međusobno povezani. Jednom kada se svi pikseli povežu, svakoj se komponenti pridruži određen graylevel ili boja koja je predviđena za tu grupu piksela.

Ovaj algoritam radi na principu 4-connectivity ili 8-connectivity, odnosno poredi svaki piksel sa susednih 4 (gore, dole, levo, desno) ili susednih 8 piksela. U praksi, odnosno na nasem datasetu, CCL se nije pokazao kao adekvatan za resavanje problema, jer 4-connectivity nije davao precizne rezultate a 8-connectivity je imao preveliki variance.

Zato se, umesto da se posmatra rastojanje izmedju cvorova za ispitivanje egzistencije ivice u grafu, pretpostavi unapred da su svi cvorovi u grafu povezani na neki nacin (nema izolovanih cvorova). U ovom slucaju nije moguće primeniti CCL, te se koristi MCL algoritam (Markov cluster algorithm).

Markov cluster algorithm kreće od grafa sa ogromnim brojem cvorova sa ciljem da se taj broj cvorova smanji. Funkcionise na principu matematickog bootstrapovanja, odnosno od postojećeg grafa kreira novi isključivo koriscenjem informacija iz ulaznog seta podataka.

Ovaj algoritam deterministički izracunava verovatnoće proizvoljne setnje (random walk) kroz graf, koristeći dva operatora (ekspanzije i inflacije). Ekspanzija se matematicki modeluje kao normalan proizvod stohastickih matrica, a u praksi zeli da pocevsi od prvog cvora prosiri skup cvorova koji posmatramo. Inflacija je matematicki modelovana kao Hadamardov proizvod, odnosno gde se svaki element mnozi sa njemu odgovarajucem u drugoj matrici (obe matrice moraju biti kvadratne i istih dimenzija), a ovo se radi sa ciljem da se odrede verovatnoće da se iz jednog cvora grafa (jednog piksela) stigne u drugi. Svakoј grani je dodeljena verovatnoca da se iz jednog cvora grane stigne u drugi. Uzimaju se samo verovatnoće od najvećeg znacaja a postojeće grane sa malim verovatnocama se brisu. Na ovaj nacin se ispostavlja da se graf jako precizno podeli u komponente nakon određenog broja koraka, a u praksi ovo obuhvati sve piksele slicnih boja, odnosno sa malom razlikom u vrednosti.