

Interaktivna igra križaljke sudoku

Branko Radoš¹

¹Faculty of Electrical Engineering and Computing,
University of Zagreb, Croatia

branko.rados@fer.hr

June 4, 2018

Abstract. Ovaj članak obrađuje temu interaktivnog igranja slagalice sudoku. Gdje su prikazani različiti segmenti i njihovi problemi prilikom pronalaženja rješenja križaljke Sudoku iz slik početnog stanja Sudoku križaljke.

Keywords. Sudoku, Konvolucijska neuronska mreža, Canny, Sobel, Hough.

1 Uvod

Kao što vjerojatno već znate, rješavanje sudokua uključuje popunjavanje 9x9 polja tako da svaki redak, stupac i glavni 3x3 blok sadrži brojeve od 1 do 9. Na početku popunjena su samo neka od polja, poznatija kao tragovi. Za razliku od drugih numeričkih slagalica, u sudokuu nema aritmetike: elementi Sudoku mreže mogu također biti slova abecede ili bilo koji drugi simboli.

2 Obrada, analiza i razumjevanje slike

2.1 Uklanjanje šuma

Svaka snimljena fotografija podložna je utjecaju okoline. Gdje okolina može malo "uništiti" sliku ili ju kompletno "uništiti".

Šum na slici se uklanja na mnoge načine. Jedna od jednostavnih, a opet tako moćna je tehnikom morfološkog zatvaranja, a zatim morfološkog otvaranja. Morfološko otvaranje uklanja male objekte iz prvog plana slike i postavlja ih u pozadinu, dok morfološko zatvaranje male sitne rupe iz prvog plana, mijenjajući male otoke pozadine u prvi plan. Ove tehnike mogu se još koristiti za pronalazak određenih oblika na slici. Morfološko otvaranje može se iskoristiti za pronalazak

stvar u koje specifična struktura stana.

2.2 Sobelov operator

Kako bi se detektirale linije odnosno rubovi korišten je Sobelov operator.

Sobelova maska za x-os:

$$G_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \quad (1)$$

Sobelova maska za y-os:

$$G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad (2)$$

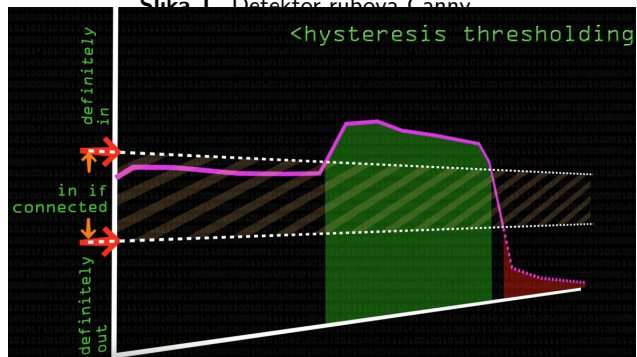
Sobel je diskretni diferencijabilni operator, koji računa aproksimaciju gradijenta funkcije intenziteta slike. U svakoj točki slike rezultat Sobelovog operatora je ili gradijentni vektor ili norma vektora. Sobelov operator temelji se na konvoluciji slike s malim, separabilnim, cjelobrojnim filterom u horizontalnom ili vertikalnom smjeru i zbog toga nije računalno zahtjevan. Izlaz Sobelovog operatora je korišten kao ulaz u canny detektor rubova.

2.3 Canny detektor rubova

Canny detektor rubova je operator za detekciju rubova koji koristi višeprocetni algoritam za detekciju širokog raspona rubova na slici. Canny prvo istanji sve rubove tako da su širine jedna piksel odnosno uzme maksimum ruba, a ostali dio makne (na primjer ako imamo rub u obliku Gaussa Canny će uzeti samo maksimum Gaussa, a ostatak postaviti na nulu). Canny je histerezni threshold odnosno postavlja se donja i gornja granica sve što je iznad gornje granice automatski je rub, a sve što je ispod granice nije rub. Za područja koja se nalaze u vrijednostima koje su između donje i gornje granice vrijedi pravilo ako su povezani s dijelom čija je vrijednost iznad gornje granice proglašavaju se rubom inače nisu rub. Odnosno s canny filtrom mičemo ne željene rubove (najčešće šum) koje je Sobelov operator detektirao.[1]



Slika 1. Detektor rubova Canny

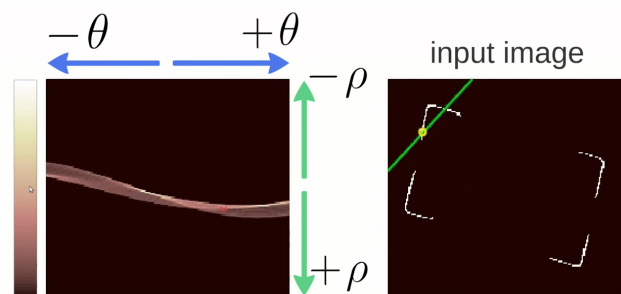


Slika 2. Detektor rubova Canny

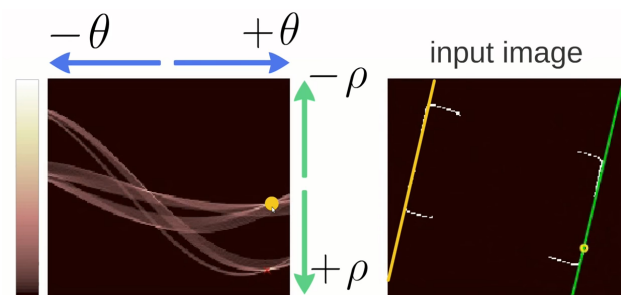
2.4 Hough-ova transformacija

Izlaz Canny-evog detektora ulazi u Hough-ovu transformaciju kako bi se detektirale linije.

Hough-ova transformacija je tehnika ekstrakcije značajki korištena u analizi slike, računalnom vidu i digitalnoj obradi slike. Svrha transformacije je pronaći nesavršene instance željenog oblika objekta na principu glasanja. Glasanje se odvija u prostoru parametara, gdje se kandidati biraju kao lokalni maksimumi u takozvanom akumulatorskom prostoru kojeg je sam algoritam konstruirao računajući Houghovu transformaciju.



Slika 3. Houghova transformacija lijevo je prostor parametara, desno je slika nakon primjene Sobelovog operatora



Slika 4. Houghova transformacija lijevo je prostor parametara, desno je slika nakon primjene Sobelovog operatora

2.5 Dimenzije polja

Radi utjecaja preostalog šuma sve linije nisu uspješno detektirane zbog toga se koristio median udaljenosti između linija kako bi se uspješno odredila veličina

svakog polja.

2.6 Analiza popunjenosti polja

Kada su poznate sve informacije o izgledu igračkog polja sudokua. Polje se dijeli na 81 igračke polje i predaje se neuronskim mrežama na klasifikaciju. Prva neuronska mreža određuje popunjenost polja odnosno imali li broja u polju ili ne dok druga neuronska mreža određuje broj u polju. Prva neuronska mreža trenirana je na praznim poljima, MNIST-u i degeneriranom MNIST-u dok je druga trenirana na MNIST-u i degeneriranom MNISTU. Iz tih razloga potrebno je igračka polja transformirati na veličinu 28×28 . Izlaz neuronske mreže koristiti se za popunjavanje matrice koja se predaje algoritmu za rješavanje sudokua.

Prva neuronska mreža ima strukturu :

- convolution2dLayer(3,128, 'Padding' , 'same')
- batchNormalizationLayer
- reluLayer
- maxPooling2dLayer(2, 'Stride', 2)
- convolution2dLayer(3,256, 'Padding' , 'same')
- batchNormalizationLayer
- reluLayer
- maxPooling2dLayer(2, 'Stride', 2)
- convolution2dLayer(3,256, 'Padding' , 'same')
- batchNormalizationLayer
- reluLayer
- convolution2dLayer(3,256, 'Padding' , 'same')
- batchNormalizationLayer
- reluLayer
- convolution2dLayer(3,512, 'Padding' , 'same')
- batchNormalizationLayer
- reluLayer

- fullyConnectedLayer(256)
- reluLayer
- fullyConnectedLayer(126)
- reluLayer
- fullyConnectedLayer(2)
- softmaxLayer
- classificationLayer

Druga neuronska mreža ima strukturu :

- convolution2dLayer(3,128, 'Padding' , 'same')
- batchNormalizationLayer
- reluLayer
- maxPooling2dLayer(2, 'Stride', 2)
- convolution2dLayer(3,256, 'Padding' , 'same')
- batchNormalizationLayer
- reluLayer
- maxPooling2dLayer(2, 'Stride', 2)
- convolution2dLayer(3,256, 'Padding' , 'same')
- batchNormalizationLayer
- reluLayer
- maxPooling2dLayer(2, 'Stride', 2)
- convolution2dLayer(3,256, 'Padding' , 'same')
- batchNormalizationLayer
- reluLayer
- maxPooling2dLayer(2, 'Stride', 2)
- fullyConnectedLayer(256)
- reluLayer
- fullyConnectedLayer(126)
- reluLayer
- fullyConnectedLayer(11)
- softmaxLayer
- classificationLayer

2.7 Rješavanje sudokua s algoritmom rekurzivnog praćenja unatrag

Kako bi se pokazao rad algoritma korištena je jednostavniji slučaj u kojem je mreža 4x4 s glavnim blokovima 2x2 [2]. Slagalica ovakvog oblika zove se "Shidoku" umjesto Sudoku zato što "Shi" na japanskom znači "četiri".

Slika 5 prikazuje prvi primjer Shidoku slagalice. Slike 6, 7, 8 prikazuju postupak rješavanja. Na slici 6 prikazani su potencijalna rješenja odnosno kandidati. Na slici su prikazani kao mali brojevi. Redak 2 sadrži sadrži 3, a stupac jedan sadrži 1, stoga su kandidati za poziciju (2,1) 2 i 4. Četiri polja sadrže samo po jednog kandidata. Takva polja zovu se singletoni. Ovaj primjer može se jednostavno riješiti popunjavajući polja singletonovima. Na slici 7 umetnut je jedan od singletonova i ponovno su izračunati kandidati. Na slici 8 umetnuti su preostali singletonovi koji ujedno predstavljaju kompletno rješenje slagalice.

1			
		3	
	2		
			4

Slika 5. Shidoku je Sudoku na 4x4 mreži.

Teži primjer prikazan je u nastavku.

Pošto nema niti jednog singletona u slagalici, koristimo rekurzivnu metodu praćenja unazad. Odaberemo jedno polje i uvjetno umetnemo jednog od kandidata. Odabir kandidata je vršen prema numeričkom redoslijedu. Zato je u polje (2,1) umetnut broj 3, tvoreći novu slagalicu. Nakon toga imamo rekurzivni poziv programa.

1	3 4	2 4	2
2 4	4	3	1 2
3 4	2	1	1 3
3	1 3	1 2	4

Slika 6. Kandidati. Crveni kandidati su singletoni.

1	3 4	2 4	2
2 4	4	3	1 2
4	2	1	1 3
3	1	1 2	4

Slika 7. Umetanje singletona 3 i ponovno računanje kandidata.

Nova slagalica je sada lagana, rezultat je prikazana na slici 12. Međutim, rješenje ovisi o odabiru kojeg smo izvršili prije rekurzivnog poziva. Drugačiji odabir može rezultirati drugačijim rješenjem. Za ovaj jednostavan slučaj kada je početno stanje dijagonalna matrica, postoje dva rješenja, koja su rješenje prikazano na slici 12 i transponirano rješenje sa slike 12. Pošto rješenje nije jedinstveno, početno stanje prikazano na slici 9 nije valjano.

1	3	4	2
2	4	3	1
4	2	1	3
3	1	2	4

Slika 8. Umetanje ostalih singletona za dovršavanje slagalice.

1			
3	2		
		3	
			4

Slika 11. Uvjetno umetanje 3 za kreiranje nove slagalice. Onda unazadno praćenje.

1			
	2		
		3	
			4

Slika 9. Shidoku je Sudoku na 4x4 mreži.

1	3 4	2 4	2 3
3 4	2	1 4	1 3
2 4	1 4	3	1 2
2 3	1 3	1 2	4

Slika 10. Kandidati. Nema niti jednog singletona.

1	4	2	3
3	2	4	1
4	1	3	2
2	3	1	4

Slika 12. Rezultat slagalice. Rješenje nije jedinstveno, prikazano rješenje transponirano također je rješenje.

3 Zaključak

U radu je prikazan put obrade, analize i razumijevanja slike od samog ulaza do generiranja samo rješenja Sudoku-a. U samom algoritmu uzete su dosta općenite pretpostavke i razmatranja ulazne slike:

- Ulazna slika sadrži kompletnu slagalicu
- Kut pod kojim je uslikana slagalica nije uvijek pravi
- Uspješno je detektirano barem dvije uzastopne linije u horizontalnom, kao i u vertikalnom smjeru.

Rješavanje Sudoku križaljke osnovno školskom djetetu ne predstavlja neki izazov naspram računalu koji mora

provesti niz računskih obrada kako bi riješio na oko jednostavan problem.

Literatura

1. **Corke, P. (2013).** *Robotics, Vision and Control: Fundamental Algorithms in MATLAB.* Springer Publishing Company, Incorporated, 1st edition. ISBN 3642201431, 9783642201431.
2. **Moler, C.** Solving sudoku with matlab. <https://www.mathworks.com/company/newsletters/articles/solving-sudoku-with-matlab.html>. Accessed: 2018-06-03.