

DIGITALIZACIJA IN VEKTORIZACIJA ČRTNIH RISB VELIKIH DIMENZIJ

INFORMATICA 4/91

A. Dedić

R. Murn

D. Peček

INSTITUT JOŽEF STEFAN

LJUBLJANA

Keywords: image analysis and processing,
pattern recognition, vectorization, GIS

Odsek za računalništvo in informatiko

Programska orodja in aplikacije, ki tečejo na sodobnih računalniških sistemih omogočajo zajem, uporabo in upravljanje prostorsko porazdeljenih podatkov. Zaradi velike količine podatkov, mora biti vnos hiter in učinkovit. Avtomatski vnos grafičnih podatkov nam omogočajo postopki za natančno digitalizacijo in vektorizacijo grafičnih predlog, ki so narisane na papirju in njemu sorodnih medijih. V delu so proučeni postopki za digitalizacijo in vektorizacijo črtnih risb velikih dimenzij. Na osnovi ovrednotenja teh postopkov in zahtev uporabnikov so izdelani osnovni pogoji za kakovostno in hitro digitalizacijo grafičnih predlog in transformacijo rasterske slike v vektorski zapis. To so: natančnost postopka, uporabnost rezultata in hitrost. Izdelan je lasten postopek za digitalizacijo in vektorizacijo, ki optimalno izpolnjuje navedene pogoje.

DIGITALIZATION AND VECTORIZATION OF LARGE SCALE DRAWINGS AND MAPS

Modern programing tools and applications running on up-to-date computer systems enable efficient managing of spatial distributed data. Due to enormous amount of image data, graphic data input must be extremely fast and reliable. In the paper raster to vector conversion techniques are discussed. There are two basic requests: usefulness of the result, and the speed of data acquisition. Dedicated raster to vector converter is shown.

Uvod

Človeški rod je že od pradavnine iz različnih razlogov zbiral podatke o svojem okolju. Težišče tega zbiranja so bili v glavnem geografski in topografski podatki. V 19. stoletju so bila prvič izvedena triangulacijska dela kot osnova novih kartografskih projektov. Prav tako se začne razvijati sistem zemljiškega katastra. V našem stoletju je stekel razvoj še mnogo hitreje.

Nasprotno od hitrega razvoja tehnik pridobivanja podatkov o prostoru je tehnika hranjenja teh podatkov napredovala le počasi. Od kamnitih plošč, papirusa je prišel papir, ki se je obdržal do današnjih dni. V zadnjih dveh dekadah se za shranjevanje in obdelavo podatkov nezadržno širi uporaba računalniške tehnologije. Razviti so računalniški sistemi, ki zagotavljajo učinkovito zbiranje, vodenje, povezovanje in vrednotenje geometrijskih podatkov o okolju. Take sisteme imenujemo Geografski Informacijski Sistemi (GIS).

Zajemanje podatkov je časovno in finančno najzahtevnejša komponenta GIS-a. Po nekaterih podatkih je delež vnosa

podatkov 80 odstotkov celotne investicije. Poleg vseh ostalih zahtev zajemanja, kot sta hitrost in enostavnost, je bistveni pogoj pri zajemanju ohranjanje izvirne natančnosti podatkov. Sodobna tehnologija omogoča relativno enostavno in hitro zajemanje analognih grafičnih predlog (kart, map) z pomočjo posebnih linijskih kamer, in pretvorbo v digitalno obliko. Rezultat se shrani v računalnik v rasterski obliki. Članek obravnava pretvorbo rasterskih slik v vektorsko obliko.

Definicija problema

Digitalno sliko, lahko v rasterskem zapisu enostavno in hitro prikažemo na računalniškem zaslonu. Vendar ta oblika zapisa ne nosi informacije o vsebini grafične predloga.

Vektorski zapis (vektorska slika) grafičnih podatkov je primerna za računalniško obdelavo in nosi strnjen opis vseh informacij o grafični predlogi, ki jo ponazarja. Zato je

potrebno razviti ustrezen računalniški postopek za transformacijo rasterskih podatkov v vektorske. Transformacija mora izpolnjevati nekatere zahteve. Najvažnejše je, da se ohrani izvorna natančnost podatkov, saj je njihova vsebina informacija o metriki prostora, ki je zakonsko predpisana.

Računalniški postopek mora omogočati enostaven, hiter in natančen vnos podatkov iz grafičnih predlog v računalniško podprt geografski informacijski sistem [1,2,3].

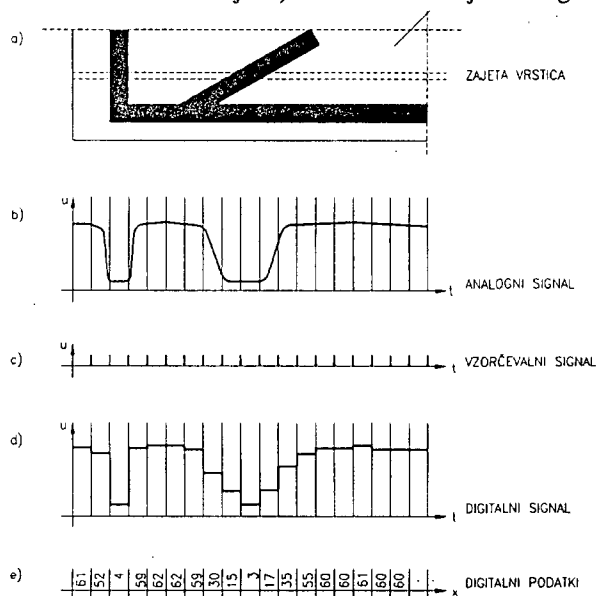
Celotni postopek je razdeljen na dva zaporedna, smiselno in programsko ločena dela:

- a) digitalizacija črtnih risb velikih dimenzij,
- b) vektorizacija dobljenih podatkov.

Prvi del vsebuje opis problematike digitalizacije grafičnih predlog velikih dimenzij. V drugem delu so detaljno analizirane zahteve za transformacije podatkov iz rasterskega v vektorski zapis.

Digitalizacija črtnih risb

Pod pojmom "digitalizacija" razumemo proces zajemanja črtnih risb z ustrežno kamero (linijsko - enodimenzijsko ali ploskovno - dvodimenzijsko) in transformacijo analognih



Slika 1. Analogni signal in digitalni podatki

signalov (podatkov) v digitalno obliko. Taki transformirani digitalni podatki so primerni za obdelavo in za hranjenje v digitalnem računalniku. Transformacija zveznih analognih signalov v analogne vzorce se izvaja z elektronskim vezjem v kameri. Analogne signale vzorčimo v predpisani frekvenci in določamo digitalno vrednost vzorcev v odvisnosti od amplitude (Slika 1). Število digitalnih razredov je ponavadi

in-*N*-ta potenca števila 2. Značilne vrednosti za *N* so 1, 4, 6, 8, Najvišji amplitudi odgovarja najvišji digitalni razred in najnižji amplitudi, najnižji razred. Vse vmesne amplitude se zaokrožijo navzdol na ustrezni razred.

Za črtne risbe večjih dimenzij (standardni formati A2, A1, A0 in večji) je značilno, da z današnjo tehnologijo ni možna enostavna in cenovno dostopna digitalizacija z dvodimenzijsko kamero zaradi premajhne ločljivosti ustreznih senzorjev. Za te namene se je uveljavila uporaba ustrezno prirejenih linijskih kamer. Na tržišču je uveljavljen izraz "SKENER"; imenovali ga bomo "linijski digitalizator". Z linijskim digitalizatorjem lahko digitaliziramo grafične predloge, ki so omejene z fizično širino linijskega senzorja (ponavadi več kot en meter) in teoretično neomejene dolžine. Dolžina je omejena z velikostjo pomnilnika v računalniku v katerega se zapisujejo podatki iz digitalizatorja. Važen podatek za linijski digitalizator je ločljivost zajemanja. Standardne ločljivosti so 100, 200, 300 in 400 točk na inčo (*dots per inch* - DPI). Novejši digitalizatorji dosežejo do 600 DPI.

Za potrebe aplikacij katerih cilj je zapis segmentov iz grafične predloge v matematični obliki (*vektORIZACIJA*), je najbolj ugodno, če je slika zapisana v dveh nivojih: belo za ozadje slike in črno za črte na sliki. Črtne risbe, ki so najbolj zanimive in so vplivale na razvoj magistrske naloge so:

- katasterske mape standardnih meril,
- geografske mape različnih meril,
- geodetske, gozdarske, elektrovodne, komunalne, in druge mape,
- inženirski načrti vseh vrst in
- druge grafične predloge, ki so izrisane z črtami kakršne koli oblike.

Črtne risbe so največkrat risane v eni barvi na kontrastni podlagi. Obstajajo tudi take, ki so risane v več barvah, vendar na različnih slojih - prozornih folijah, kar obravnavamo kot več različnih, enobarvnih predlog. Za starejše tipe grafičnih predlog, ki so risane v več barvah na eni sami podlagi lahko uporabimo barvne filtre pri digitalizaciji (*barvna separacija*), tako da dobimo več različnih enobarvnih predlog. To je značilno za veljavne katasterske načrte iz obdobja Marije Terezije. Takšni načrti še zmeraj pokrivajo približno 80 procentov površine Republike Slovenije. Osnovni problem pred vsebinsko obdelavo digitalizirane slike je pravi izbor algoritmov (izbor uveljavljenih in/ali izdelava novih). Kompleksno sliko je namreč potrebno razbiti na eno ali več homogenih površin, ki opisujejo posamezne logične nivoje. Ta problem imenujemo segmentizacije ali razčlenjevanje slike. Popularna metoda segmentizacije slike je "rezanje" slike na enem sivinskem nivoju (*image thresholding*). Ta metoda je računsko bolj enostavna in hitrejša kot druge metode za segmentizacijo (odkrivanje robov na sivinski sliki z različnimi operatorji ali razne tehnike, ki slonijo na odkrivanju homogenih področij na sliki).

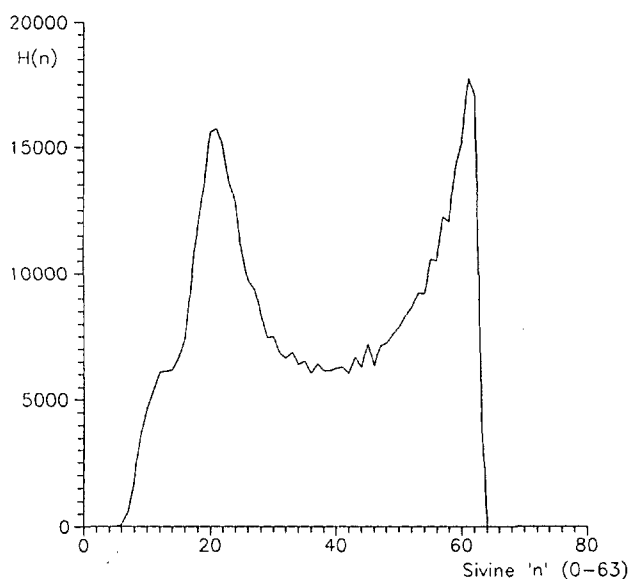
Metoda rezanja slike je v bistvu metoda iskanja optimalne *PRAGOVNE* vrednosti. Tako na enostaven način dobimo uporabno dvonivojsko sliko. V splošnem primeru je ta metoda dosti bolj zahtevna, če so črte na sliki, ki jih je treba segmentizirati v različnih sivinah (npr. N svin) in je zato potrebno poiskati več ustreznih pragovnih vrednosti ($N - 1$).

Za segmentizacijo kakovostnih črtnih risb je zadostna in potrebna predstavitev z dvema nivoji, *črna - belo*. V teh primerih kaže tudi histogram sivinskih vrednosti izrazito dvopasovno obliko: pas, ki odgovarja točkam na črtah in pas, ki odgovarja točkam ozadja (Slika 2). Na tak način lahko sliko f segmentiziramo na dva dela tako, da upoštevamo razmerje sivinskih vrednosti $G(x,y)$ proti eni sivini T . Vrednost T - ja leži med maksimumom svin črt in maksimumom svin ozadja.

Ta enostavni kriterij za vse točke slike f se da predstaviti kot:

- če je $G(x,y) > T$ potem je točka iz skupine črt ali skupine 1,
- sicer je točka iz skupine točk ozadja ali skupine 2.

Vrednost T imenujemo pragovna vrednost (*threshold*), postopek pretvorbe slike iz večsivinskega zapisa v dvonivojski pa binarizacija.



Slika 2. Histogram kontrastne črtne risbe kaže izrazito dvopasovnost

Digitalizacija in binarizacija pripravi sliko za nadaljno obdelavo, v našem primeru vektorizacijo. Kakovost digitalizacije in binarizacije vpliva na rezultat vektorizacije črtne risbe (ne glede na uporabljeno metodo vektorizacije) [4,5].

Linijski digitalizator omogoča nastavitve pragovne vrednosti za binarizacijo grafične predloge. Kakovost tako

dobljenih binarnih slik je zadovoljiva samo za del novejših črtnih risb, ki so risane na posebnih predlogah s stabilnimi, kvalitetnimi peresi. Dobljeni rezultati za ostale črtne risbe so slabši. Razlog je pragovna vrednost, ki je nastavljena globalno, enotno za celo grafično predlogo, ne glede na spreminjanje odtenkov sivine narisanih črt po celi sliki in spreminjanje odtenkov barve papirja kot ozadja [6].

Na slabšanje rezultatov vplivajo:

- slaba kvaliteta papirja, ki s staranjem spreminja barvo,
- če je papir izrabljen, izdrsan (izdrgnjen), umazan na kakršen koli način, lepljen z neustreznimi lepili (selotejp trakovi),
- če je risan z slabimi peresi z nestabilnim črnilom, ki s časom spreminja barvo, ali s preveč vodenim črnilom.

Te slabosti se kažejo na digitalni, globalno binarizirani sliki kot:

- črte so neenakomerno debele, na enem mestu pretanke, drugje predebele,
- prekinjene črte ali jih sploh ni,
- na mestih, kjer je gostota črt zelo velika (naselja, križišča), so črte odebeljene, ponekod zlite v črne ploskve.

Vse metode iskanja pragovne vrednosti za digitalno, večsivinsko sliko lahko v splošnem razdelimo na dve skupini:

- metode iskanja globalne pragovne vrednosti (*global threshold selection*) in
- metode iskanja spremenljive (lokalne, dinamične, spremenljive) pragovne vrednosti (*variable threshold selection*).

S prvimi določimo za celo sliko enotno pragovno vrednost. Ostale metode določajo pravila, po katerih se pragovna vrednost spreminja po sliki od točke do točke.

Lastna metoda (variabilno načelo [7])

Pragovna vrednost T_n se za vsako točko n računa kot:

P_2	P_3	P_4
P_1	P_0	P_5
P_8	P_7	P_6

Slika 3. Oznake indeksov sosednjih točk

$$T_n = [C_n \sigma_n + \mu_n], \quad n = 1, 2, 3, \dots, N$$

N je število vseh točk na sliki f , μ_n je srednja vrednost tekoče točke in njenih osmih sosedov (Slika 3).

$$\mu_n = \sum_i P_i, \quad (i = 0, 1, \dots, 8)$$

σ_n je standardno odstopanje devetih točk od srednje vrednosti μ_n :

$$\sigma_n = \left[\sum_i \frac{(P_i - \mu_n)^2}{9} \right]^{(1/2)}, \quad (i = 0, 1, \dots, 8)$$

Koeficijentu C_n smo dali ime "pragovni koeficijent" in je odvisen od histograma cele slike $H(n)$:

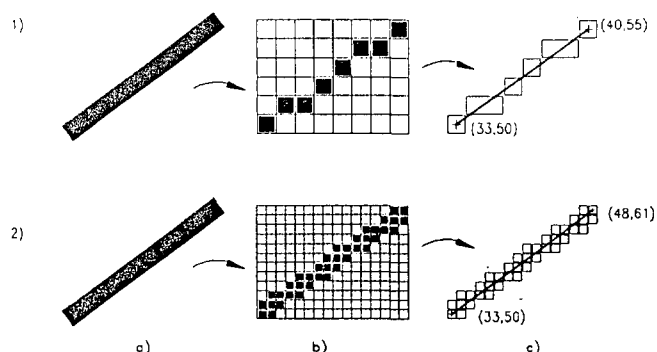
$$C_n = \frac{k [H(n)]}{N}$$

Koeficijentu C_n smo dali ime "pragovni koeficijent" in je odvisen od histograma cele slike $H(n)$:

$H(n)$ je število točk, ki imajo sivinsko vrednost n na celi sliki f . Vrednost k je konstanta. Dodali smo jo za izboljšanje rezultatov in je odvisna od kvalitete načrta. Konstanta k se določa z izkušnjo in ima vrednost med 1 za boljše predloge in 0.65 za slabše.

Vektorizacija črtnih risb

Črtna risba, ki je digitalizirana in binarizirana, je pripravljena za glavni del obdelave, vektorizacijo. Pod pojmom "vektorizacija" razumemo transformacijo zapisa črte iz rasterskega v vektorski zapis in jo imenujemo tudi "raster-vektor" transformacija. Rasterski zapis črt je digitalizirana in binarizirana slika teh črt (Slika 4). Omejimo se samo na pravokotno rastersko mrežo, ki je zgrajena iz kvadratnih rasterskih celic (obstoja tudi heksagonalna, itd).



Slika 4. Digitalizirana črta v dveh ločljivostih 1 in 2:

- (a) Narisana črta na papirju,
- (b) rasterska slika črte v različnih ločljivostih,
- (c) vektorski zapis črte.

V našem primeru vektorski zapis definiramo kot matematičen opis črte z dvema končnima točkama, koordinatami v dvodimenzionalnem, $x - y$ prostoru. Z inverzno preslikavo "vektor - raster" se pogosto srečujemo v računalniškem okolju, ko je potrebno črte, ki so zapisane v vektorski obliki, izrisati na neko rastersko podlago (računalniški monitorji, matrični tiskalniki itn). Matematične črte nimajo debeline, pri rasterski predstavitvi pa jim jo moramo prirediti - najmanj eno slikovno celico.

Idealno rastersko sliko neke črtne risbe dobimo, če so znani vsi vektorji na tej risbi in jih izrišemo na rasterski podlagi z omenjenimi algoritmi. Preslikava nazaj, iz idealne rasterske slike v vektorsko, je enolična samo, če je znan algoritem po katerem je potekala preslikava v prvi smeri. Razlog je v tem, da lahko skozi rastersko črto, debelo eno ali več celic, povlečemo neskončno mnogo vektorjev.

Digitalizirana in binarizirana slika črtne risbe je rasterska slika, ki ni idealna. Položaj vektorjev na risbi seveda ni znan, saj je bistvo vektorizacije prav odkrivanje le-teh. Torej, ta preslikava ni enolična. Iz ene rasterske slike lahko dobimo neskončno mnogo različnih, medseboj podobnih vektorskih slik. Da bi to število zmanjšali, oziroma preslikavo čim bolj poenotili, lahko postavimo različne zahteve, ki jih morajo izpolnjevati dobljene vektorske slike.

Večina črtnih risb, ki jih obdelujemo, vsebuje metriko o prostoru, ki ga predstavljajo. Informacije o tem prostoru (ali večina le-teh) so v narisanih črtah na papirju. To pomeni, da bo vsaka, še tako majhna napaka pri transformaciji pomenila popačenje v novi vektorski predstavitvi prostora. Primer tega je katasterski načrt, na katerem so narisane zemljiščne parcele. Davek na zemljo je odvisen od površine parcele. Površina večjih parcel se vidno spreminja pri še tako majhnem premiku dolgih mej - črt.

Velja da originalna črtna risba ne vsebuje idealne metrike o prostoru temveč samo približek. Zato je splošna zahteva pri vektorizaciji, da se dobljena vektorska slika čim bolj ujema z rastersko.

Kakovost vektorizacije je odvisna od uporabljene metode. Preučili smo in testirali vse dostopne metode iz literature [8-11] in kot rezultat tega in zahtev uporabnikov postavili lastne zahteve za pogoje vektorizacije. Na osnovi teh zahtev in upoštevajoč delne rešitve iz omenjene literature, smo izdelal lastno metodo za vektorizacijo. V nadaljevanju najprej sledi pregled nekaterih bolj zanimivih in pogostejše uporabljenih metod vektorizacije.

Pregled postopkov

Velika večina postopkov vektorizacije črtnih risb uporablja kot prvi korak (ali predobdelavo) različne metode za tanjšanje črt na rasterski sliki na debelino ene rasterske celice (Tabela 1). Razlog za tako široko uporabo te predobdelave je v tem, da so postopki za nadaljno obdelavo teh črt, debelih samo eno celico, že razviti in obdelani znotraj postopkov, ki so namenjeni za druge različne potrebe.

Večinoma so to različne metode iskanja in izločevanja črt in črtnih oblik kot so:

- operatorji za iskanje robov objektov na sliki,
- konturni filtri,
- sledenje roba objekta na sliki (*border tracking*),
- algoritmi za odkrivanje linijskih segmentov na sliki itn.

Zato bi lahko v splošnem vse postopke za vektorizacijo razvrstili v dve skupini:

- skupina A: tiste, ki kot prvi korak pri obdelavi uporabljajo različne metode za tanjšanje rasterskih črt
- skupina B: tiste, ki slonijo na vektorizaciji rasterskih črt originalne širine.

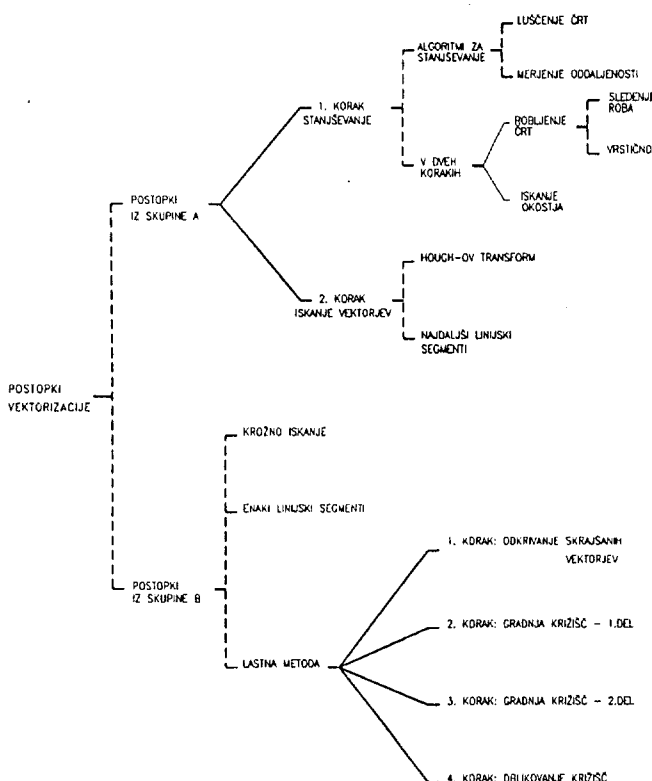


Tabela 1. Razvrstitev postopkov za vektorizacijo

Merila za vrednotenje postopkov za vektorizacijo

Da bi lahko ovrednotili postopke vektorizacije, moramo najprej določiti merila. Na prvem mestu so to zahteve ljudi iz stroke, ki bodo uporabljali programske rešitve za vektorizacije.

Za vrednotenje smo izbrali tri osnovna merila:

1. natančnost metode,

2. uporabnost dobljenega rezultata,

3. časovna zahtevnost postopka.

Po prvem merilu vrednotimo natančnost dobljenih rezultatov (ujemanje vektorskih črt z rastersko sliko). Minimalna zahtevana natančnost je, da vektor leži znotraj točk črte na rasterski sliki.

Drugo merilo so zahteve in želje uporabnikov. Naše izkušnje so pokazale, da zaenkrat ni mogoče implementirati univerzalnega algoritma, katerega rezultat bi bil sprejemljiv za različne vrste uporabnikov. To pomeni, da je potrebno vektorizacijske algoritme prilagoditi vsakemu uporabniku posebej. To se največkrat da doseči z nastavljanjem različnih parametrov vektorizacijskega procesa.

Pomembno je tudi tretje merilo. Zaradi izredno velikega števila risb, ki jih je treba obdelati, morajo biti postopki prilagojeni. Posebej še, če naj ti algoritmu tečejo na cenениh računalnikih tipa *IBM PC*. Izpolnjevanje vseh treh pogojev je potreben in zadosten pogoj za kakovostno vektorizacijo.

Lastni postopek za vektorizacijo

Osnovno vodilo in smernice pri izdelavi lastnega sistema za vektorizacijo so bili trije osnovni pogoji. Prva ugotovitev je bila, da je potrebno celotni postopek razbiti v več enostavnejših korakov. Iskanje postopkov s katerimi bi v enem prehodu dobili uporaben končen rezultat ne daje zahtevanih rezultatov.

1. KORAK: Odkrivanje skrajšanih vektorjev

Osnovni problem vektorizacije je ogromna količina podatkov, ki jih je treba obdelati. Na primer: za črtno risbo dimenzije $800 \times 700 \text{ mm}$ (katasterski načrt), ki je digitalizirana, z ločljivostjo 600 DPI dobimo 320 milijonov črno belih točk ali 40 Mby (Mega zlogov) podatkov. Z zapisom podatkov v kodah tekočih dolžin dosežemo faktor kompresije približno 5 , to je 8 Mby . To je še vedno veliko, vendar bistveno lažje za obdelavo.

Postopek vektorizacije mora zaradi ogromne količine podatkov teči na komprimirani sliki. Zapis v obliki kod tekočih dolžin je zelo ugoden, ker je komprimiran in ima obliko iz katere se hitro ugotovi potek rasterskih črt. Zato je tak zapis rasterske slike osnova za lasten postopek vektorizacije.

Obdelava rasterske slike mora teči zaporedno, od enega konca k drugemu. To je v skladu z zapisom slike. Kakršna koli obdelava v naključni smeri (recimo: sledenje rasterski črti po sliki) je časovno zelo zahtevna. Zato smo za prvi korak izbrali obdelavo slike v eni smeri: od zgoraj navzdol (vrstico za vrstico). S tako obdelavo občutno zmanjšamo število posegov računalnika do zunanjih enot na katerih je slika zapisana (*trdi disk*). Za zapis rasterske slike v kodah tekočih dolžin je karakteristično, da so za vsako vrstico zapisane dolžine nizov izmenično črnih ali belih točk. Tem podatkom rečemo *črna* ali *bela polja*.

Ko pri obdelavi slike naletimo na prvo vrstico, ki ni prazna, ~~zapišemo~~ v posebna polja podatkov, vsako črno polje posebej. V teh poljih vodimo evidenco o začetih črtah (Slika 5.A). Za vsako naslednjo vrstico navzdol (tekoča vrstica) testiramo vsako začeto črto posebej, če se nadaljuje z enim izmed črnih polj v tej vrstici.

Če se črta nadaljuje, potem testiramo naslednje pogoje po vrsti (če je eden izmed pogojev izpolnjen, ne testiramo naprej):

a) če je črno polje s katerim se črta nadaljuje tudi nadaljevanje za eno ali več sosednjih začetih črtah (Slika 5.D),

b) če v tekoči vrstici obstoja še eno ali več črnih polj, ki so nadaljevanje testirane začete vrstice (Slika 5.C),

Če so testi a) in b) negativni potem:

c) povlečemo ravno črto od sredine prvega črnega polja testirane črte do sredine zadnjega črnega polja (za katero je črta podaljšana). Testiramo, če eno izmed črnih polj vmes odstopa od narisane črte za več, kot je predpisano odstopanje r_0 . Obstajata dve možnosti:

c1) če je narisana črta znotraj vseh prejšnjih črnih polj, potem v polje podatkovno tekoči črti dopišemo zadnje črno polje (Slika 5.E),

c2) če celotna črta ni znotraj vseh črnih polj, potem začeto črto končamo s črnim poljem v prejšnji vrstici in začnemo novo s črnim poljem v tekoči vrstici (Slika 5.F).

Če je pogoj a) pozitiven (to je slučaj, če se testirana črta deli v eno ali več novih črt), potem se tekoča črta konča (kot črta od njenega začetka do črnega polja prejšnje vrstice) in se začne evidenca o eni ali več črtah z začetkom v tekoči vrstici.

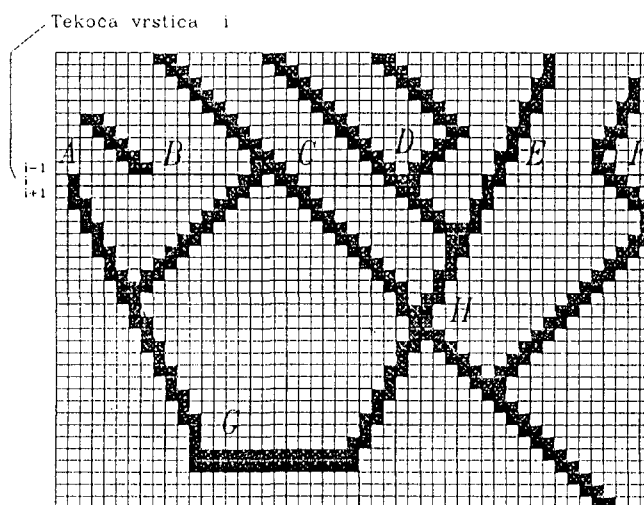
V primeru b) (slučaj, da se ena ali več prej začetih črt združuje v eno novo) se v prejšnji vrstici konča ena ali več začetih črt in se začne evidenca o novi črti z začetkom v tekoči vrstici.

Če se ena izmed črt ne nadaljuje v tekoči vrstici (niso izpolnjeni pogoji za nadaljevanje), se le-ta konča v prejšnji vrstici (Slika 5.B). Polja podatkov v katerih smo vodili evidenco o končanih črtah sprosti sproščamo.

Koordinate vseh končanih črt (vektorje) shranimo na posebno datoteko. Pri shranjevanju se lahko rešimo nepotrebnih kratkih vektorjev, ki večinoma predstavljajo vektorizirane napise in simbole na grafičnih predlogah. Če je vektor krajši od najkrajše vnaprej določene dolžine r_1 , potem ga ne shranimo (Slika 5.H).

V vsakem križišču po prvem koraku, ostanejo luknje (Slika 5). V prvem koraku križišč ne gradimo. To je v bistvu osnovna ideja, na kateri je bil zasnovan celoten postopek:

Za uspešno vektorizacijo črtnih risb, ki vsebujejo metriko o prostoru je ključnega pomena natančno identificirati vektorje narisanih črt, oziroma koordinate dveh končnih točk vsake posamezne črte (vsaka črta je enolično definirana s svojima končnima točkama). Če je črta samostojna (prosta na obeh koncih), potem končne točke ni težko določiti. Problem je, da so večinoma ti dve točki v vozliščih treh ali večih črt (črte se združujejo ali razdružujejo v vozliščih). Zelo težko je natančno določiti koordinato posameznega vozlišča oziroma eno končno točko vseh črt, ki so združene v vozlišču. Na osnovi tega smo prišli do večkorakne rešitve: v prvem koraku vse črte skrajšamo za del, ki leži v vozlišču, (črte vektoriziramo samo do vozlišča). S postopki v naslednjih korakih še oblikujemo vsa vozlišča.



Slika 5. Prvi korak vektorizacije

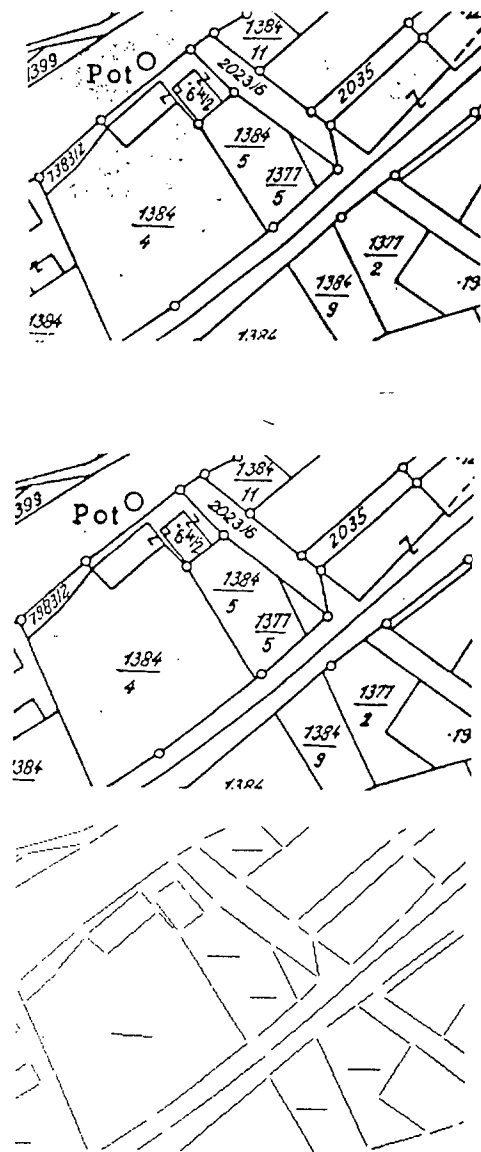
S testiranjem navedenih pogojev v prvem koraku: a), b), in c) hitro in enostavno odkrivamo vozlišča. Postopek v prvem koraku je učinkovit, ima pa pomanjkljivost. Preciznost odkrivanja skrajšanih vektorjev je odvisna od kota pod katerim je narisana črta. Za kote od 45 do 135 stopinj (navpične črte) je natančnost zelo velika. Za kote od -45 do 45 stopinj (horizontalne črte) natančnost ni zadovoljiva (Slika 5.G).

Za rešitev tega problema smo našli enostavno in učinkovito rešitev. Rasterska slika, ki je zapisana s kodami tekočih dolžin, se prepiše v popolnoma enako novo sliko; samo je zapis kod za novo sliko drugačen. Pravila za zapis kod ostanejo ista, vendar se kode ne pišejo več po vrsticah, temveč po stolpcih, z leve proti desni. Slika vsebuje kode, ki so zarotirane za 90 stopinj.

Za novo, rotirano sliko isti postopek ponovimo. Vektorje vertikalnih črt originalne slike bomo dobili, če upoštevamo samo črte, ki so pod kotom od 45 do 135 stopinj (prvi del prvega koraka - obdelava nerotirane slike) (Slika 6). Vektorje horizontalnih črt dobimo če upoštevamo samo črte, ki

so pod kotom od 45 do 135 stopinj, pri obdelavi rotirane slike (to so vektorji črt, ki so na originalni sliki pod kotom od - 5 do 45 stopinj) (Slika 6). Celotna vektorizirana slika V je unija teh dveh podslik $V_{(45,135)}$ in $V_{(-45,45)}$. Prvi korak lahko ponazorimo z dvema preslikavama iz rasterskega R v vektorski prostor V :

$$\begin{aligned} R &\longrightarrow V_{(45, 135)} \\ R &\longrightarrow V_{(-45, 45)} \\ V &= V_{(45, 135)} \cup V_{(-45, 45)} \end{aligned}$$



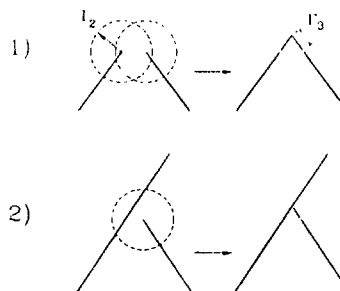
Slika 6. a)kopija dela črtne risbe
b)binarizirana slika
c)rezultat po prvem prehodu

2. KORAK: Izgraditev vozlišč, prvi del

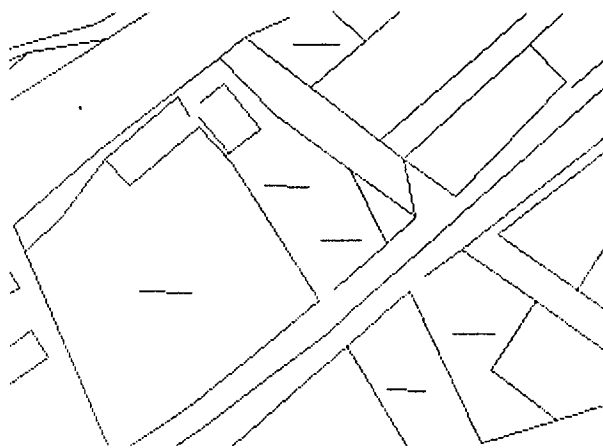
Za drugi korak je značilno da obdelava poteka samo na vektorski sliki, dobljeni v prvem koraku. Obdelava vektorske slike je bistveno hitrejša od obdelave rasterske slike. Po prvem koraku ostane še zmeraj problem za črte, ki so približno pod kotom 45 oz. 135 stopinj. Za te kote obstaja

možnost podvojitve vektorjev zato, ker je lahko npr. črta, ki je pod kotom 45 stopinjah vektorizirana na nerotirani in tudi na rotirani sliki. Zato v drugem koraku vse vektorje, ki so približno pod kotoma 45 in 135 stopinj na nerotirani sliki, testiramo če so enaki (s predpisanim odstopanjem) na rotirani sliki. V slučaju obstoja takih parov odstranimo po eno črto iz para. S tem je predobdelava vektorjev v drugem koraku končana.

V glavnem delu drugega koraka testiramo vsak vektor, če obstaja v njegovi vnaprej določeni bližini, r_2 konec drugih vektorjev. To je večinoma slučaj na mestih preskočenih vozlišč, združevanja dveh ali treh črt in redkeje štirih ali več. V primeru pozitivnega testa, podaljšamo odkrite vektorje do presečišča. Če je podaljšek krajši od naslednje pragovne vrednosti r_3 , vpišemo nov vektor na mesto starega. Na ta način zgradimo večino vozlišč. Na Sliki 7. so prikazane različne možnosti podaljševanja vektorjev v vozlišča.



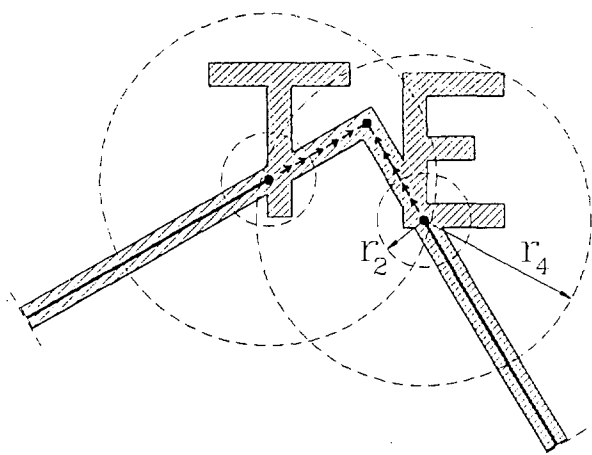
Slika 7. Pogoji podaljšanja vektorjev v drugem koraku



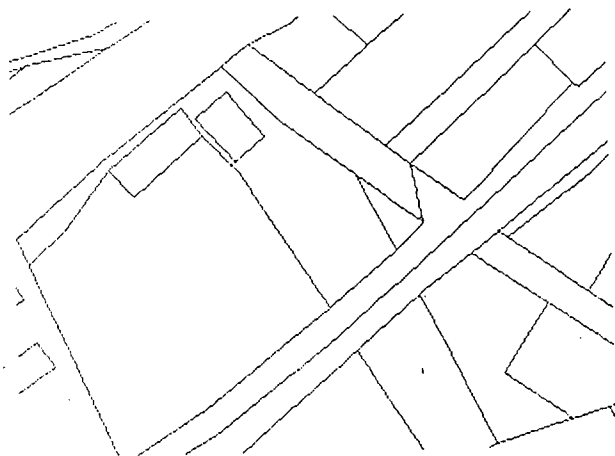
Slika 8. Rezultat po drugem prehodu

3. KORAK: Gradnja vozlišč, drugi del

Po drugem koraku lahko ostane manjši del vozlišč nezaključen. Ta primer je največkrat zaradi relativno majhne pragovne vrednosti r_2 . V drugem koraku r_2 ne smemo povečati, ker bi lahko prišlo do nepravile interpretacije rasterske slike.



Slika 9. Pogoji za podaljšanje vektorjev v tretjem koraku



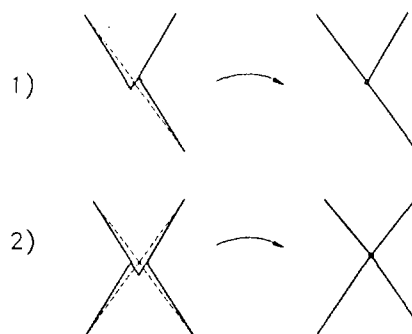
Slika 10. Rezultat po tretjem koraku

Zato v tretjem prehodu kombinirano testiramo vektorsko in rastersko sliko, vendar samo lokalno. Določimo novo pragovno vrednost r_4 , ki je ponavadi dvakrat večja od r_2 . Testiramo preostale nepodaljšane vektorje pod istimi pogoji kot v drugem prehodu (samo z r_4 namesto r_2). Če najdemo v področju r_4 še eno ali več nepodaljšanih vektorjev, potem začasno podaljšamo vse te vektorje do presečišča. Na rasterski sliki testiramo, če so novi vektorji podaljšani preko rasterskih črt (Slika 9). Če je test pozitiven, lahko vpišemo nove podaljšane vektorje na mesto starih.

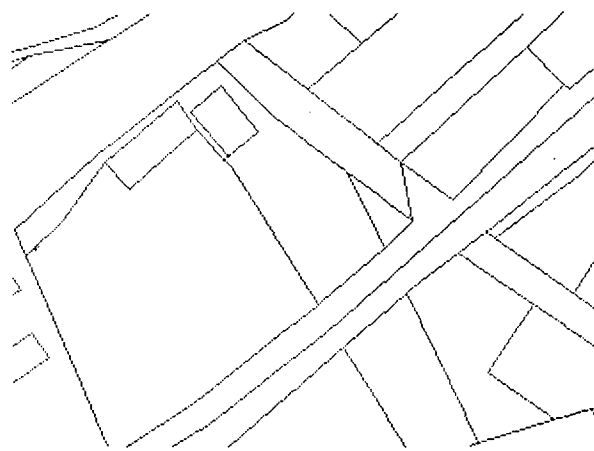
4. KORAK: Oblikovanje vozlišč

Vozlišča, ki združujejo tri ali več črt in so bila zgrajena v drugem in tretjem prehodu, največkrat nimajo popolnoma pravilne oblike. Možne oblike nepravilnosti so povečano prikazane na Sliki 11. Te nepravilnosti so zelo majhne, registrirajo pa jih informacijski sistemi, ki uporabljajo vektorski zapis kot vhodne podatke. Zato v četrtem prehodu preverimo vsa vozlišča in jih ustrezno oblikujemo, kot je to

prikazano na Sliki 11. Po četrtem koraku je postopek vektorizacije končan.



Slika 11. Oblikovanje vozlišč v četrtem koraku



Slika 12. Rezultat po četrtem koraku

Zaključek

Geografski Informacijski Sistemi (GIS) so računalniška orodja, ki zagotavljajo učinkovito vodenje, povezovanje in vrednotenje geometrijskih podatkov o okolju. Osnovni problem za izgradnjo GIS-ov je vnos podatkov v računalnik.

Večina podatkov o okolju je še vedno na papirju. Obstajajo različni ročni postopki vnosa teh podatkov v računalnik. V nalogi so preučeni postopki za digitalizacijo in vektorizacijo črtnih risb velikih dimenzij, katerih struktura opisuje metriko prostora. Ti postopki so avtomatizirani. Vnos podatkov je tako lažji in hitrejši.

Postopek vektorizacije črtnih risb je v veliki meri odvisen od vrste in vsebine črtne risbe, ki jo obdelujemo. V različnih okoljih so postavljeni različni standardi. V članku je predstavljena lastna metoda digitalizacije in vektorizacije, ki je zasnovana na zahtevah uporabnikov slovenskega prostora.

Postopek digitalizacije je priprava rasterske slike za nadaljno obdelavo s postopkom vektorizacije. Vektorizacija se izvaja v štirih zaporednih korakih. Izpolnjuje tri osnovne pogoje za uspešnost: natančnost rezultatov,

sprejemljivost dobljenih rezultatov in hitrost postopka. Na posebej izdelani testni sliki je bil preverjen naš postopek za prenos grafične informacije iz predloge v vektorski zapis. Testiranje je pokazalo, da naša metoda izpolnjuje vse tri pogoje za uspešno vektorizacijo.

OPOMBA: Podani postopki predstavljajo avtorsko delo in jih ni dovoljeno reproducirati. Prav tako avtorji ne odgovarjajo za škodo, ki bi lahko nastala ob poiskusu implementacije le teh.

Literatura

- [1] A. Dedić, R. Murn, D. Peček, Postopki za obdelavo slik, *YUROB'89*, 1989, pp. 4.63-4.67.
- [2] A. Dedić, R. Murn, D. Peček, Optični zajem in digitalizacija grafičnih predlog večjih dimenzij, *YUROB'90*, 1990, pp. 3.38-3.42
- [3] A. Dedić, R. Murn, D. Peček, Digitalization of Large Area Drawings and Maps, *Melecon'91, IEEE Mediterranean Conf*, 1991, pp. 1260-1263.
- [4] M.T.Musavi at al., A vision Based Method to Automate Map Processing, *Pattern Recognition*, 1988, pp. 319-326.
- [5] M. Ejiri at al., Automatic recognition of design and maps, *Proc. 7th Int Conf. Pattern Recognition*, pp. 1296-1305 (1987).
- [6] N. Otsu at al, A threshold selection method from gray-level histograms. *IEEE Trans. Systems Man Cybernet.* 9, 1979, 62-66.
- [7] Y. Nakagawa, Some experiments on variable thresholding, *Pattern recognition* 11, 1979, 191-204
- [8] R. Esplid at al., A Raster-to-Vector-Conversion System Producing High Quality Geometric Entities, *The 6th Scand. Conf. on Pattern Rec.*, 1989, 19-22
- [9] S. Suzuki, Graph-Based Vectorization Method for Line Patterns, *Int. Conf. on Pattern Recognition*, 1988, pp. 616-621.
- [10] S. Suzuki at al., Automatic line drawing Recognition of Large-Scale Maps, *Optical Eng.* 7, 1987, pp. 642-649
- [11] K. Ramachandran, Coding Method for Vector Representation of Engineering Drawings, *Proceedings of the IEEE* 68, 1980, pp. 813-816.