



Marković, V., Jakovljević, Ž., Budak, I.¹⁾

PREPOZNAVANJE CILINDARA I RAVNI U TRODIMENZIONIM OBLACIMA TAČAKA²⁾

Rezime

U radu se predlaže metod za prepoznavanje cilindara i ravni u nestruktuiranim oblacima tačaka. Predloženi proces prepoznavanja se može podeliti u tri osnovne faze. Prvu fazu predstavlja automatska segmentacija širenjem regiona počev od jedne karakteristične tačke. Kriterijumi širenja regiona zasnovani su na osobinama singularnosti informacione matrice sistema kao i pripadnosti tačaka površi čiji su parametri estimirani metodom najmanjih kvadrata. Druga faza algoritma se odnosi na grupisanje presegmentiranih oblasti i estimaciju parametara prepoznatih cilindara i ravni. Dobre performanse ovoj fazi obezbeđuje upotreba unapređenog algoritma robusnog prepoznavanja cilindara iz oblaka tačaka kao i uvođenje procesa precizne estimacije parametara ravni. Na samom kraju, odnosno u trećoj fazi procesa upotrebom predloženog algoritma vrši se ponovna obrada celokupnog polaznog oblaka tačaka u cilju ekstrakcije prepoznatih primitiva i obezbeđivanja preciznih krajnjih rezultata. Predloženi metod je pre svega namenjen prepoznavanju cilindara i ravni u oblacima tačaka koji reprezentuju određene mašinske delove, pa je u skladu sa tim i eksperimentalno verifikovan na većem broju odgovarajućih sintetizovanih oblaka.

Ključne reči: reverzno inženjerstvo, obrada oblaka tačaka, segmentacija površi, prepoznavanje cilindara, prepoznavanje ravni

1. UVOD

Savremena industrija sve više spoznaje benefite od upotrebe tehnika reverznog inženjerstva i kao takva se trudi da ih integriše u svoje tradicionalne procese stvarajući time nove koncepte. Razvoj i projektovanje proizvoda verovatno je prva oblast u kojoj je pomenuta integracija u velikoj meri i ostvarena. Danas većina CAD programskih paketa sadrži module koji podržavaju reverzno inženjerstvo geometrijskih modela. Ovde se pre svega misli na kreiranje kompjuterskog 3D modela samo na bazi realnog objekta tačnije njegovog oblaka tačaka dobijenog različitim visokorezolutnim 3D skenerima. Da bi se oblak tačaka preveo u 3D model mora se proći kroz nekoliko faza [1]: 1) preprocesiranje oblaka tačaka, 2) prepoznavanje geometrijskih primitiva i 3) formiranje kompjuterskog modela skeniranog objekta na bazi dobijenih informacija iz prethodnih faza. Preprocesiranje oblaka tačaka može biti raznovrsno i zavisi od metoda koji se dalje primenjuje, ali svakako najčešće se sastoji od generisanja 3D trougaonih mreža. 3D mreže trouglova u nekim slučajevima predstavljaju estetski dobre reprezentacije skeniranih objekata i kao takve mogu biti zadovoljavajući izlaz iz procesa reverznog inženjerstva. Međutim, ovakve mreže ne mogu dati dovoljno dobre informacije o obliku i strukturi skeniranog objekta (geometrijskim primitivima od kojih se sastoje) i zbog toga se samo na osnovu njih, a bez dalje obrade, ne mogu kreirati precizni 3D modeli. Takođe, potrebno je napomenuti da se u nekim slučajevima korak generisanja 3D mreža izostavlja i da se u daljim koracima procesa vrši isključivo obrada samo početnog (ili redukovano) oblaka tačaka.

Prepoznavanje geometrijskih primitiva iz oblaka tačaka može se podeliti na dva koraka [1]: 1) segmentaciju oblaka tačaka na oblasti raličite po svojim geometrijskim svojstvima, i 2) prepoznavanje i

¹⁾ M.Sc Veljko Marković, Univerzitet u Beogradu, Mašinski fakultet, (markovicveljko@yahoo.com), prof. dr Živana Jakovljević, Univerzitet u Beogradu, Mašinski fakultet, (zjakovljevic@mas.bg.ac.rs), prof. dr Igor Budak, Univerzitet u Novom Sadu, Fakultet tehničkih nauka, (budaki@uns.ac.rs)

²⁾ U okviru ovog rada saopštavaju se rezultati istraživanja koja se sprovode na projektima pod evidencionim brojevima TR35004 i TR35020 koji finansijski podržava Ministarstvo prosvete, nauke i tehnološkog razvoja Republike Srbije.

estimaciju parametara geometrijskih primitiva koje reprezentuju segmentirane oblasti. Segmentacija oblaka spada u najvažnije, ali i najzahtevnije korake obrade oblaka tačaka [2]. Ona se po pravilu izvodi uz asistenciju korisnika CAD softvera, pri čemu je potrebno da korisnik vizuelno prepozna geometrijski različite regione i obeleži određeni broj tačaka na njima. Nakon toga, softver može samostalno izvršiti estimaciju parametara primitiva. Na osnovu navedenog može se zaključiti da je razvoj procesa automatske segmentacije oblaka tačaka od esencijalnog značaja za buduće unapređenje prepoznavanja geometrijskih primitiva primenom CAD softvera kao i za korišćenje tehnika 3D digitalizacije u automatizaciji proizvodnih procesa. Efikasni algoritmi za ovu svrhu su još uvek u razvoju i nisu dostigli dovoljan nivo za praktičnu primenu u realnom vremenu. Sa druge strane, ukoliko bismo se ograničili na oblast mašinstva, uvideli bismo da većina delova poseduje strukturu koja je zasnovana pretežno na ravnima i cilindrima. Iz navedenih razloga u fokusu ovog rada nalazi se koncept automatskog prepoznavanja cilindara i ravni iz nestruktuiranih oblaka tačaka koji ujedno uključuje i njihovu automatsku segmentaciju.

Prepoznavanje geometrijskih primitiva najčešće je zasnovano na segmentaciji oblaka tačaka koja se može svrstati u grupu segmentacija zasnovanih na rastu (širenju) regiona [3]. Algoritmi ovog tipa u [2, 4, 5] koriste jednu izabranu polaznu karakterističnu tačku od koje se region širi dodavanjem tačaka u okolini, a po nekom geometrijskom kriterijumu. Izbor polazne tačke može biti manuelni [2, 4] ili automatski [5]. Sa druge strane, kriterijumi za širenje regiona uglavnom su zasnovani na parametrima diferencijalne geometrije čime se dovodi do toga da metod segmentacije bude veoma osetljiv na izbor polazne tačke. Takodje treba napomenuti da je za osnovu za širenje regiona moguće izabrati i grupu tačaka umesto samo jedne tačke [6].

U ovom radu se predlaže metod za automatsku segmentaciju i prepoznavanje cilindara i ravni iz nestruktuiranih oblaka kao i estimaciju njihovih parametara. Predložen metod se sastoji iz tri osnovna koraka (algoritma): 1) segmentacija oblaka koja u zavisnosti od kompleksnosti objekta za rezultat može imati segmentaciju jednog te istog geometrijskog primitiva u veći broj regiona (presegmentacija), 2) grupisanje presegmentiranih oblasti i estimacija parametara površi, i 3) izdvajanje cilindara i ravni iz oblaka. Predloženi korak segmentacije može se svrstati u pomenutu grupu segmentacija sa širenjem regiona počev od jedne izabrane karakteristične tačke, a po određenim kriterijumima koji će u radu biti objašnjeni. Grupisanje presegmentiranih oblasti izvodi se nakon segmentacije pri čemu se vrši i prepoznavanje površi i estimacija njihovih parametara. U poslednjem koraku vrši se izdvajanje prepoznatih cilindara i ravni iz oblaka i formiranje konačnog skupa rezultata.

Nastavak ovog rada ima sledeću strukturu po poglavljima. Drugo poglavlje odnosi se na metod segmentacije oblaka tačaka. U trećem poglavlju predstavlja se grupisanje presegmentiranih površi kao i estimacija njihovih parametara. Poglavlje 4 objašnjava konačni korak izdvajanja cilindara i ravni iz oblaka tačaka. Rezultati implementacije predloženog metoda na primeru tri sintetizovana oblaka koja reprezentuju određene mašinske delove biće prikazani u poglavlju 5, dok se u završnom, šestom poglavlju daju zaključne napomene.

2. SEGMENTACIJA OBLAKA

Kao što je već naglašeno, u procesu automatskog prepoznavanja geometrijskih primitiva iz oblaka tačaka najznačajniji korak predstavlja segmentacija oblaka na klastere (grupe tačaka) koje reprezentuju određene površi različite po svojim geometrijskim osobinama. U našem slučaju jasno je da oblak tačaka treba podeliti na oblasti koje reprezentuju određene geometrijske primitive od kojih se površine skeniranog objekta sastoje. Metod segmentacije koji koristimo u ovom radu kreira klastere širenjem regiona počev od jedne tačke po određenim kriterijumima u dva koraka. Inicijalno širenje regiona se odvija na osnovu osobina singularnosti informacione matrice dobijene tokom procesa regresije parametara površi na osnovu skupa tačaka, dok se konačno širenje regiona obavlja proverom pripadnosti tačaka inicijalnom klasteru čiji su parametri estimirani metodom najmanjih kvadrata.

Na osnovu ranijih istraživanja [7,8,9] zaključeno je da je dobar način formiranja inicijalnih grupa tačaka širenje regiona od jedne tačke dodavanjem okolnih tačaka (jedne po jedne) prateći osobine singularnosti informacione matrice. Naime, ukoliko su tačke na površi uzorkovane bez ikakve greške i suma odnosno egzaktno pripadaju određenoj površi, informaciona matrica sistema tačaka će biti singularna. Ovo sa druge strane znači da ukoliko skup tačaka dobro aproksimira neku površ (što je i slučaj kod oblaka tačaka dobijenih skeniranjem) onda će informaciona matrica biti bliska singularnoj, odnosno njen recipročni faktor uslovljenosti će biti blizak (ali ne i jednak) nuli. Upravo ovakava osobina je i iskorišćena u ovom radu za inicijalno širenje regiona. Potrebno je dodati i da ovakav pristup omogućava da segmentacija ne bude osetljiva na izbor početne tačke, što je od velikog značaja za konačne rezultate.

Oblik informacione matrice sistema tačaka koji bi bio pogodan za segmentaciju zavisi od vrste geometrijskih primitiva čije reprezentativne klustere želimo da izdvojimo. Imajući u vidu rezultate iz [7,8,9], a i u skladu sa činjenicom da cilindri pripadaju grupi površi drugog reda (grupi kvadraka), informacionu matricu sistema definišaćemo počevši od opšte jednačine kvadraka iskazane u obliku:

$$a_1x^2 + a_2y^2 + a_3z^2 + a_4xy + a_5yz + a_6xz + a_7x + a_8y + a_9z + a_{10} = 0 \quad (1)$$

gde $[x \ y \ z]$ predstavljaju koordinate tačke na površi, dok su $a_i, i=1,\dots,10$ parametri površi. Ovu jednačinu možemo takođe predstaviti i u matricnom obliku:

$$\mathbf{x} \cdot \mathbf{a} = 0 \quad (2)$$

gde je $\mathbf{x}=[x^2 \ y^2 \ z^2 \ xy \ yz \ xz \ x \ y \ z \ 1]$, dok $\mathbf{a}=[a_1 \ a_2 \ \dots \ a_{10}]^T$ predstavlja vektor parametara površi.

Interesantno je primetiti da matematički zapis (1) sa određenim koeficijentima jednakim nuli predstavlja opštu jednačinu ravni. Na osnovu navedenog se može zaključiti da se informaciona matrica dobijena na osnovu jednačine (1) može iskoristiti i za segmentaciju ravni iz oblaka tačaka.

Matrica regresionog modela sa n tačaka je formata $n \times 10$ i može se definisati kao:

$$\mathbf{D}=[x_i^2 \ y_i^2 \ z_i^2 \ x_iy_i \ y_iz_i \ x_iz_i \ x_i \ y_i \ z_i \ 1], i=1,\dots,n \quad (3)$$

dok je tražena informaciona matrica:

$$\mathbf{S} = \mathbf{D}^T \mathbf{D} \quad (4)$$

Upravo se osobine singularnosti ovako postavljene informacione matrice koriste za početno širenje regiona u ovom radu što će kasnije biti detaljnije i objašnjeno. Sa druge strane, estimacija parametara izdvojenog klastera metodom najmanjih kvadrata može se izvršiti rešavanjem problema minimizacije:

$$\begin{aligned} \min \|\mathbf{D}\mathbf{a}\|^2 \\ \text{na osnovu } \mathbf{a}^T \mathbf{C}\mathbf{a} = 1 \end{aligned} \quad (5)$$

gde je \mathbf{C} matrica formata 10×10 čiji je jedini element različit od nule $C(10,10)=1$.

Postavljeni problem dalje vodi jednačini:

$$\mathbf{S}\mathbf{a} = \lambda \mathbf{C}\mathbf{a} \quad (6)$$

gde λ predstavlja uvedeni Lagranžov množilac.

Rešenje problema je jednoznačno i predstavlja sopstveni vektor koji odgovara najmanjoj sopstvenoj vrednosti informacione matrice \mathbf{S} [10]. Vrednosti ovog sopstvenog vektora odgovaraju traženim estimiranim parametrima površi koji se dalje koriste za dopunsko širenje regiona.

Početak segmentacije predstavlja izbor prve početne tačke regiona (engl. *seed point*) za koju se bira tačka sa najvećom z koordinatom. Potom se izvodi obrada oblaka algoritmom sa k najbližih suseda, tzv. knn (engl. *k-nearest neighbours*) algoritmom odnosno vrši se izračunavanje Euklidovih distanci od izabrane tačke do svih ostalih tačaka oblaka i tačke se sortiraju od najbliže ka najdaljoj. Nakon toga, pristupa se dodavanju jedne po jedne tačke (redom počev od najbliže) u klaster i u svakoj iteraciji se proverava da li je vrednost recipročnog faktora uslovljenosti informacione matrice sistema manja od unapred definisane veoma niske pozitivne vrednosti praga za inicijalnu segmentaciju. Ukoliko je vrednost manja, tačka se dodaje klasteru i širenje regiona se nastavlja daljim iteracijama. Dolazak do tačke koja ne zadovoljava postavljeni uslov ujedno znači i prestanak iteracija čime se na izlazu dobija klaster sačinjen od do tog trenutka izdvojenih tačaka. U ovom segmentu algoritma potrebno je ograničiti minimalni dozvoljeni broj izdvojenih tačaka za dalje procesiranje, odnosno potrebno je sprečiti dalju upotreba previše malih grupa tačaka za estimaciju parametara površi. U našem slučaju ograničenje je fiksno postavljeno na 20 tačaka i ukoliko se desi da se izdvoji manji broj tačaka odabir se poništava i ponovo se vrši novo širenje regiona počev od prve sledeće tačke.

Nakon formiranja dovoljno velikog inicijalnog klastera, pristupa se dopunskom širenju regiona. U tu svrhu prvo se vrši estimacija parametara površi na osnovu do tada izdvojenih tačaka. Zatim se proverava pripadnost estimiranoj površi svih preostalih tačaka oblaka čime se vrši izdvajanje tačaka koje na osnovu unapred definisanog praga zadovoljavaju estimiranu jedačinu kao uslov pripadnosti površi. Ovim postupkom dolazi se do kompletnog izdvojenog klastera. Da bi se u ovom slučaju sprečilo generisanje previše malih konačnih klastera (koji nisu pogodni za estimaciju parametara površi u narednoj fazi algoritma), potrebno je ograničiti i njihovu minimalnu veličinu. U našem slučaju ovo ograničenje je postavljeno procentualno i iznosi 0.1% veličine polaznog oblaka. Ukoliko su izdvojeni klasteri manji od zadate veličine, oni se prosto ne uzimaju u razmatranje u daljim fazama algoritma.

Nakon formiranja jednog odgovarajućeg klastera, algoritam nastavlja sa novim iteracijama. Uzima se prva sledeća dostupna tačka za početnu tačku narednog regiona i procedura se ponavlja sve dok se ne izdvoje sve tačke oblaka. Pseudo kod predloženog algoritma segmentacije prikazan je na slici 1.

Opisani algoritam za segmentaciju u svojoj osnovi predstavlja kombinaciju algoritama iz [8, 9]. Naime, algoritam iz [8] je namenjen segmentaciji površi iz oblaka tačaka dobijenih skeniranjem rotacionih delova. Ovi oblaci tačaka se na jednostavan način mogu struktuirati tako da za određivanje najbližih suseda nije neophodan knn algoritam. S druge strane, kako bi se smanjio efekat presegmentacije regiona, u okviru [9] estimacija parametara vršena je na osnovu procedure za direktnu estimaciju parametara elipsoida iz [10] i uvedene su određene strategije za povećanje inicijalnog regiona. Međutim, dalja istraživanja i razvoj algoritma za spajanje presegmentiranih regiona koji će biti opisan u poglavlju 3 pokazala su da direktna estimacija parametara elipsoida korišćenjem procedura iz [10, 11] i primena razvijenih strategija dovodi do povećanja proračunske kompleksnosti bez značajnog unapređenja konačnih rezultata prepoznavanja cilindara i ravni iz oblaka tačaka.

INPUT: x, y, z – points from the cloud
thres1, thres2 – segmentation thresholds

```

cloud=[x, y, z];
m=0; trials=0;
while trials<20 %minimal number of trails
  chose seed point;
  idx=knn(seed,cloud);
  m=m+1;
  surfaces(m).surf(1,:)=cloud(seed,:);
  for i=1:length(idx)
    surf_aux=[surfaces(m).surf; cloud(idx(i,:),:)];
    calculate rcond(S) using surf_aux
    if rcond(S)<thres1
      add cloud(idx(i,:),) to surfaces(m).surf; end
  end
  if length(surfaces(m).surf)>20
    l1=0; l2=1000;
    while l1<l2
      l1=length(surfaces(m).surf)
      calculate A
      for i=1:length(cloud)
        if distance<thres2
          add cloud(i,:) to surfaces(m).surf
        else add cloud(i, :) to new_cloud end
      end
      l2=length(surfaces(m).surf)
    end
    cloud=new_cloud;
  else trials=trials+1; end
  if length(cloud)<2 trials=21; end
end

```

OUTPUT: **surfces** – structure containing segmented regions

Slika 1. Pseudo kod predloženog algoritma segmentacije

3. GRUPISANJE I ESTIMACIJA PARAMETARA POVRŠI

Tokom početnog testiranja i uvodjenja prethodno predstavljenog algoritma segmentacije primećeno je da algoritam dobro klasifikuje i odvaja regione koji su po svojim geometrijskim karakteristikama različiti. Ovo znači da se pri primeni ovakvog procesa segmentacije neće dogoditi da algoritam izdvoji region koji sadrži delove više od jednog geometrijskog primitiva (u našem slučaju reč je o ravnima i cilindrima). Međutim, primećeno je i da je algoritam segmentacije u odredjenim uslovima sklon presegmentaciji, odnosno izdvajanju većeg broja regiona nego što je potrebno. Na ovo direktno utiče nivo šuma u oblaku, ali i izabrane vrednosti pragova segmentacije. Zbog navedene negativne osobine, za potrebe ostvarenja polazne ideje potrebno je razviti dodatni postupak grupisanja presegmentiranih regiona.

Osnovu algoritma grupisanja čini proces robusnog prepoznavanja cilindara (engl. *Robust Cylinder Fitting - RCF*) iz oblaka tačaka [12] koji je za potrebe ovog rada modifikovan i unapređen. Takođe u algoritmu egzistira i precizna tehnika estimacije parametara ravni. Kompletan postupak grupisanja segmenata biće opisan u poglavlju 3.3.

3.1 Robusno prepoznavanje cilindara iz oblaka tačaka

Metod robusnog prepoznavanja cilindara iz oblaka tačaka opisan u [12] prevashodno je razvijen za potrebe obrade oblaka tačaka dobijenog skeniranjem pokretnim laserskim skenerom na autonomnom vozilu.

Kao takav, vrši prepoznavnje i izdvajanje cilindara čija je visina znatno veća od radijusa. U našem slučaju ovakvo pojednostavljenje se ne može uzeti u obzir i zbog toga se najveći deo naših izmena ovog algoritma odnosi na povećanje sposobnosti generalizacije. U nastavku je dato detaljno pojašnjenje pomenutog unapređenog algoritma robusnog prepoznavanja cilindara.

Polazni *RCF* algoritam iz [12] sastoji se iz dve faze na osnovu kojih se dobija detaljan matematički opis prepoznatog cilindra: orijentacija, visina, koordinate centra bazne kružnice i radijus cilindra (za potrebe ovog rada neće se određivati visina cilindra). U njegovoj prvoj fazi izdvojeni deo oblaka se obrađuje poznatim algoritmom robusne analize osnovnih komponenti sistema (engl. *Robust Principal Component Analysis - RPCA*) [13]. Ovom analizom primenjenom nad 3D oblakom tačaka dobijaju se vrednosti tri ortogonalna vektora pravca ($v_i, i=2,1,0$) koji odgovaraju adekvatnim sopstvenim vrednostima kovarijante matrice sistema ($\lambda_i, i=2,1,0$, obično je $\lambda_2 \geq \lambda_1 \geq \lambda_0$). Pošto je polazno pojednostavljenje u [12] pomenuta velika razlika između visine cilindra i njegovog prečnika, unapred je poznato da vektor v_2 odgovara osi cilindra i da bazu cilindra (bazni krug i njegove parametre) treba tražiti u projekciji tačaka na ravan koju čine vektori v_1 i v_0 . U našem slučaju, gde unapred nije poznato u kojoj osnovnoj ravni treba prepoznati bazu cilindra potrebno je izvršiti obradu projekcija tačaka u sve tri ravni i izabrati pravu. Zbog željenog povećanja brzine izračunavanja i činjenice da su cilindri koji se pojavljuju u strukturama mašinskih delova pretežno orijentisani duž neke od osa koordinatnog sistema skeniranog objekta u našem metodu se ne koristi *RPCA*. Umesto njega, izdvojeni klaster se projektuje u sve tri koordinatne ravni polaznog oblaka tačaka. Svaka od projekcija se obrađuje i bira se najbolja odnosno ona u kojoj se nalazi potrebna baza cilindra.

Postupak biranja najbolje projekcije izvodi se iterativnim metodom za svaku projekciju posebno. U svakoj iteraciji se proizvoljno biraju tri tačke - $h_0=3$ (što predstavlja minimalan broj tačaka za matematički opis kružnice). Na osnovu izabranih tačaka vrši se regresija kružnice metodom najmanjih kvadrata [14] (ovakva vrsta regresije spada u metode koje minimizuju algebarsku funkciju umesto minimizacije geometrijske greške regresije). Ovim postupkom dobijaju se koordinate centra (a_0, b_0) i radijus (r_0) kružnice koje koristimo za izračunavanje zbira kvadratnih odstupanja svih tačaka projekcije:

$$\sum_{i=1}^n e_i^2 = \sum_{i=1}^n \left[\sqrt{(x_i - a_0)^2 + (y_i - b_0)^2} - r_0 \right]^2 \quad (7)$$

Nakon završetka postavljenih iteracija, bira se regresija sa najmanjim kvadratnim odstupanjem i vrši se upoređivanje podataka za sve tri projekcije. Ravan čija projekcija klastera po ovom postupku pokaže najmanje odstupanje predstavlja onu za koju se sa sigurnošću može tvrditi da sadrži projekciju bazne kružnice cilindra. Ovim izborom se automatski dobija i osa orijentacije cilindra kao vektor koji je normalan na izabranu koordinatnu ravan (**i, j** ili **k**).

Pored navedenog, svakako treba istaći i postavku za izbor broja iteracija u konkretnom slučaju. Broj iteracija I_n se određuje Monte Karlo metodom kao:

$$I_n = \frac{\log(1 - p_r)}{\log\left(1 - (1 - \varepsilon)^{h_0}\right)} \quad (8)$$

gde p_r predstavlja verovatnoću dobijanja tačne regresije, dok je ε procenat očekivanih rasutih tačaka (tačaka koje ne pripadaju ni jednoj estimiranoj kružnici). U našem slučaju je $p_r=0.999$ dok je $\varepsilon=0.5$.

Nakon određivanja adekvatne bazne ravni (i projekcije klastera u njoj) i ose cilindra, u drugoj fazi izvornog *RCF* algoritma visokom tačnošću (na osnovu znatno većeg broja tačaka od h_0) se estimiraju koordinate centra i vrednost radijusa baznog kruga cilindra. Ova faza je u našem slučaju neznatno izmenjena u odnosu na polazni *RCF* algoritam i sastoji se iz tri koraka.

- Izvršiti regresiju kružnice na osnovu nasumično izabrane tri tačke (h_0) čime se dobijaju koordinate centra (a_0, b_0) i radijus (r_0). Na osnovu njih izračunati kvadratna odstupanja za svaku tačku projekcije.
- Sortirati izračunata kvadratna odstupanja i odabrati 50% posto od ukupnog broja tačaka projekcije sa najmanjim odstupanjem - $h=n/2$.
- Na osnovu h odabranih tačaka izvršiti regresiju kružnice metodom najmanjih kvadrata na isti način kao u delu za određivanje najbolje projekcije. Izračunati zatim kvadratna odstupanja za sve tačke projekcije, sortirati ih i za h onih sa namanjim odstupanjem izračunati sumu kvadratnih odstupanja.

Ponoviti korake a-c I_n puta i na osnovu sume kvadratnih odstupanja za h najboljih tačaka (iz koraka c) odabrati najbolju regresiju čiji parametri kružnice ujedno predstavljaju i konačno estimirani centar cilindra C i njegov radijus R .

Predstavljenim prilagođenim *RCF* metodom u dve faze se od izdvojenog klastera tačaka na kraju dobijaju osa \mathbf{O} , centar C i radijus R prepoznatog cilindra.

3.2 Estimacija parametara ravni

U ovom radu za matematički opis ravni koristimo implicitni oblik jednačine:

$$ax + by + cz + d = 0 \quad (9)$$

Parametri a , b i c predstavljaju koordinate vektora normale na ravan $\mathbf{n}=[a \ b \ c]$, dok je parametar d definisan kao skalarni proizvod tog normalnog vektora i vektora položaja proizvoljne tačke na ravni.

Postupak estimacije parametara ravni (a , b , c i d) koji predlažemo je iterativan. Naime, u svakoj iteraciji iz klastera se biraju proizvoljno tri tačke P_1 , P_2 i P_3 i na osnovu njih vrši se izračunavanje normale na ravan kao:

$$\mathbf{n} = \mathbf{P}_1\mathbf{P}_2 \times \mathbf{P}_1\mathbf{P}_3 \quad (10)$$

Parametar d u tom slučaju se izračunava kao:

$$d = \mathbf{n} \cdot \mathbf{P}_1 \quad (11)$$

pri čemu je \mathbf{P}_1 vektor položaja izabrane tačke P_1 .

Zbog povećanja tačnosti estimacije parametara ravni, nakon njihovog izračunavanja u svakoj iteraciji vrši se izračunavanje rastojanja svih tačaka klastera od estimirane ravni. Rastojanje svake pojedinačne tačke $P(p_x, p_y, p_z)$ od ravni može se izračunati kao:

$$D = |D_v \cdot \mathbf{e}| \quad (12)$$

pri čemu je D_v :

$$D_v = \frac{ap_x + bp_y + cp_z + d}{\sqrt{a^2 + b^2 + c^2}} \quad (13)$$

dok je $\mathbf{e} = \mathbf{n} / |\mathbf{n}|$ jedinični vektor normale ravni.

Vrednosti rastojanja svake pojedinačne tačke od ravni se zatim upoređuju sa određenom unapred definisanom vrednošću praga i na osnovu toga se određuje ukupan broj tačaka čije su distance manje ili jednake pragu. Ovaj prag teorijski može zavisiti od kvaliteta površine dela ili šuma u oblaku mada nije uočeno da mala varijacija njegove vrednosti znatno utiče na postignute rezultate. Nakon određivanja pomenutog broja tačaka u svim iteracijama pristupa se izboru parametara ravni. Biraju se oni parametri ravni pri kojima je najveći broj tačaka klastera bio u unapred zadatim granicama udaljenosti. Treba napomenuti da je za razliku od prethodnog postupka vezanog za cilindre u ovom metodu dovoljan fiksno postavljen broj od 100 iteracija.

3.3 Algoritam grupisanja

Grupisanje presegmentiranih oblasti koje se predstavlja u ovom radu, sastoji se iz dva osnovna koraka. Prvi korak podrazumeva razdvajanje klastera na one koji reprezentuju delove ravni i ostale koji u našem slučaju reprezentuju delove cilindra. Takođe, u okviru prvog koraka, vrši se i određivanje inicijalnih parametara segmentiranih površi. Drugi korak se odnosi na konkretno grupisanje klastera na osnovu tipa i vrednosti parametara.

Razdvajanje klastera prema tipu površi se izvodi prostom selekcijom na osnovu vrednosti radijusa dobijenog nakon tretiranja klastera pomenutim modifikovanim *RCF* algoritmom (poglavlje 3.1). Ukoliko je površ ravan dobiće se značajno velika vrednost radijusa i upotrebom određenog unapred definisanog praga

lako se može izvršiti potrebna klasifikacija na klastere koji reprezentuju ravni i one koji reprezentuju cilindre.

Imajući u vidu napravljenu podelu prema tipu površi koju segmentirani klasteri predstavljaju, lako se mogu dobiti početni parametri površi koji su dalje potrebni za grupisanje. Pošto su na početku grupisanja svi klasteri već tretirani *RCF* algoritmom, za cilindre su unapred određeni parametri kao što su osa orijentacije, centar i radijus. Sa druge strane, klasteri koji su označeni kao ravni se naknadno tretiraju algoritmom iz poglavlja 3.2 čime se dobijaju estimirani parametri ravni koju oni reprezentuju.

Nakon prethodno opisanog postupka razvrstavanja segmenata prema tipu i estimacije njihovih parametara, grupisanje delova se vrši poredjenjem udaljenosti tačaka od površi za ravni, odnosno na osnovu razlika u vektorima pravaca, položajima centara i radijusima za slučaj cilindara.

4. IZDVAJANJE CILINDARA I RAVNI IZ OBLAKA

U poslednjoj fazi predloženog algoritma prepoznavanja ravni i cilindara vrši se izdvajanje prepoznatih primitiva. U postupku formiranja potpunih rezultata i kvalitetne podele polaznog oblaka ova faza se pokazala kao neizostavna. Izvodi se u dva koraka. U prvom se svi do tada segmentirani i grupisani klasteri sortiraju po veličini počev od najvećeg ka najmanjem. U sledećem koraku se svaki klaster redom obrađuje, odnosno svim tačkama (preostalog) oblaka se proverava pripadnost tom klasteru. Provera pripadnosti tačaka klasterima koji predstavljaju ravni se izvodi prostom proverom rastojanja (kao u poglavlju 3.2) i njegovim upoređivanjem sa pragom za izdvajanje ravni. Sa druge strane, provera pripadnosti klasterima koji reprezentuju cilindre vrši se izračunavanjem rastojanja tačke od omotača cilindra odnosno upoređivanjem najkraćeg (normalnog) rastojanja tačke od ose cilindra i estimirane vrednosti radijusa na osnovu odgovarajućeg praga.

Treba napomenuti da se sve tačke koje pripadnu nekom od (većih) klastera u postupku izdvajanja automatski oduzimaju narednim (manjim) klasterima bez obzira na to što je u postupku segmentacije dobijen drugačiji rezultat. Takođe, tačka koja se izdvoji i pripadne nekom od klastera u ranijoj fazi izdvajanja neće više biti dostupna ostalim klasterima – dostupni deo oblaka se smanjuje.

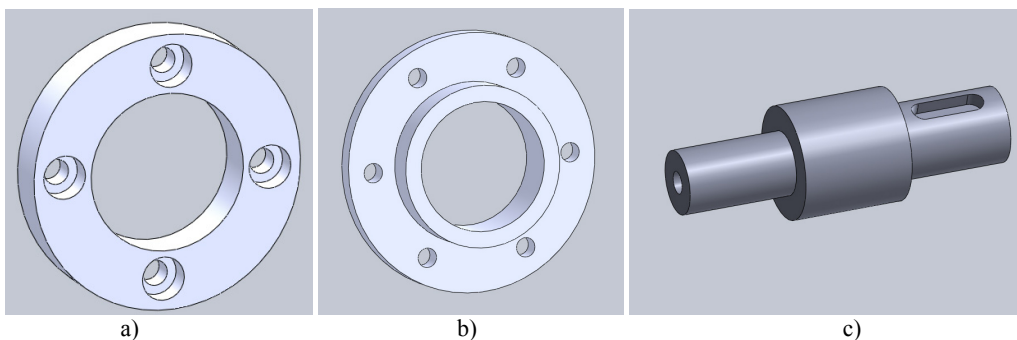
Ovakav koncept završne faze ukupnog algoritma segmentacije i prepoznavanja dovodi do dobijanja čistih regiona odnosno regiona bez suvišnih pojedinačnih tačaka na većim udaljenostima u okolini površi. Ova osobina dovodi do čiste grafičke reprezentacije rezultata što će biti moguće primetiti u poglavlju koje se bavi eksperimentalnom verifikacijom ukupnog predstavljenog metoda.

5. VERIFIKACIJA ALGORITMA PREPOZNAVANJA CILINDARA I RAVNI IZ OBLAKA TAČAKA

Nakon razrade i pojašnjenja predstavljenog algoritma ekstrakcije ravni i cilindara potrebno je isti i eksperimentalno verifikovati. U skladu sa polaznim idejama za ovu priliku iskorišćena su tri sintetizovana oblaka tačaka koja reprezentuju delove karakteristične za mašinsku industriju. U pitanju su dve prirubnice i jedno tipično vratilo čiji su CAD modeli prikazani na slici 2. U procesu sintetizovanja oblaka CAD modeli u .stl formatu visoke rezolucije su tretirani modifikovanim *z* bafer (engl. *z-buffer*) algoritmom iz tri pravca (upravno na tri osnovne ravni) u određenom neuniformnom rasteru slike. Za dobijanje tačaka oblaka korišćeni su prodori vektora posmatranja (\mathbf{i} , \mathbf{j} , \mathbf{k}) kroz površine trouglova koji su opisani .stl formatom. Nakon dobijenih prodora iz sva tri pravca vršen je presek rezultata čime su generisane konačne forme oblaka tačaka. Da bi se sintetizovani oblaci približili realnim, oni su kontaminirani šumom sa odnosom signala i šuma (engl. *Signal to Noise Ratio - SNR*) od 100dB.

Obrada oblaka tačaka predloženim algoritmom prepoznavanja cilindara i ravni iz nestruktuiranih oblaka tačaka dala je očekivano dobre rezultate. Međufazni grafički rezultati su prikazani samo za slučaj prve prirubnice (slika 3), dok su u ostalim slučajevima (slike 4 i 5) prikazani samo konačni rezultati. U tabeli 1 prikazane su vrednosti pragova korišćenih pri obradi sintetizovanih oblaka.

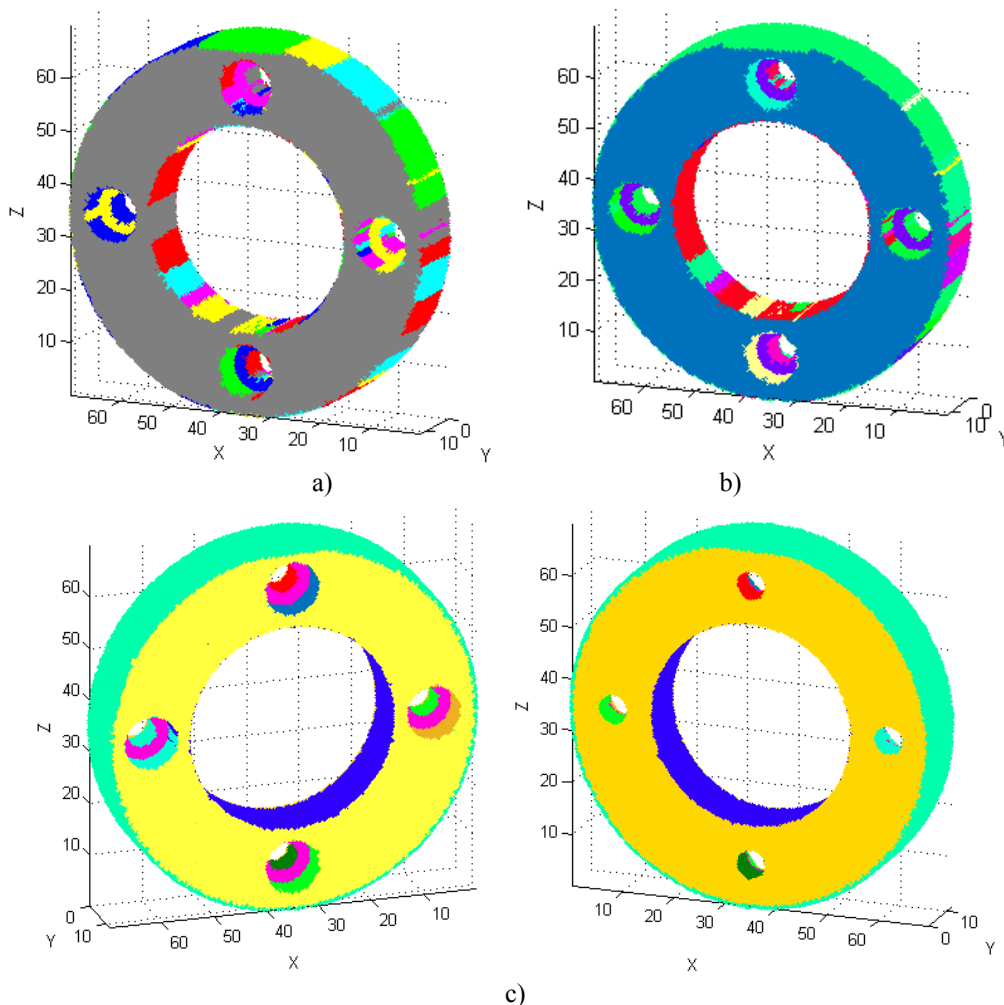
Kvalitet rezultata prepoznatih cilindara i ravni najbolje se može uočiti na osnovu vrednosti radijusa prepoznatih cilindara. Naime pri obradi svih oblaka najveća razlika između vrednosti sintetizovanih i prepoznatih cilindara iznosila je nešto manje od 0.5%. Primera radi, sintetizovani radijusi četiri vrste cilindara na prvoj prirubnici su 35, 20, 5.5 i 3.2mm, dok su odgovarajuće vrednosti prepoznatih cilindara 34.96, 19.96, 5.48 i 3.19mm.



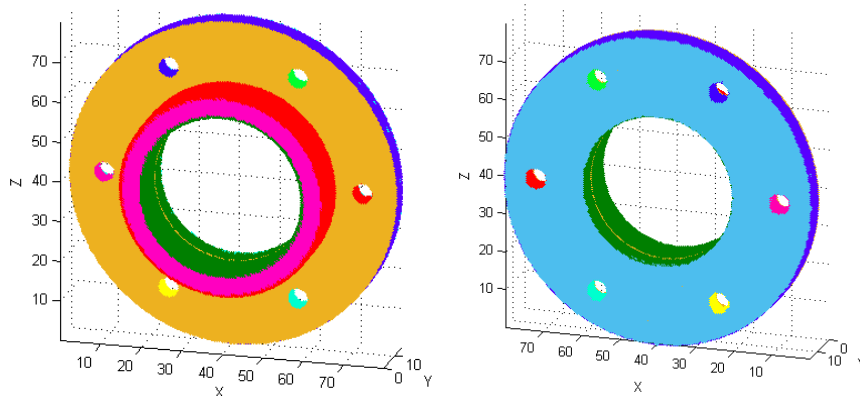
Slika 2. CAD modeli objekata korišćenih za eksperimentalnu verifikaciju: a) prirubnica 1; b) prirubnica 2; c) vratilo

Tabela 1. Vrednosti pragova korišćenih pri obradi sintetizovanih oblaka tačaka

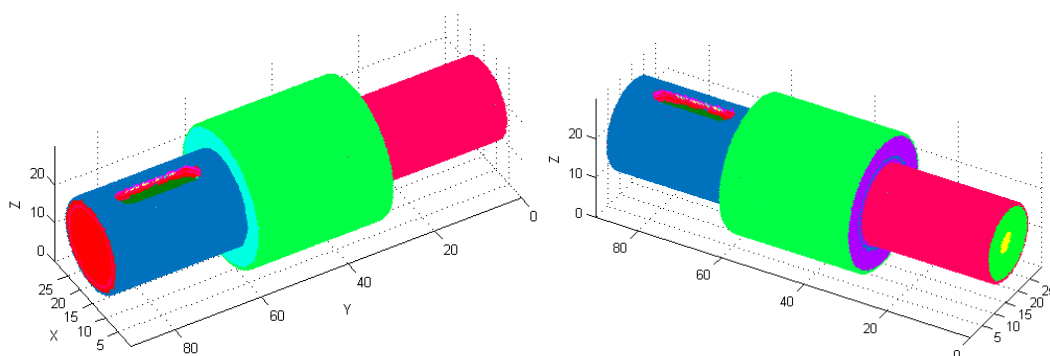
| Oblak tačaka | Pragovi za segmentaciju | | Pragovi za grupisanje | | | | | Pragovi za izdvajanje | |
|--------------|-------------------------|-------------------------|-----------------------|-----------------|-------------------------|----------------------|------------------------------|-----------------------|---------------|
| | faktor uslovljenosti | dodatno širenje regiona | parametri ravni [mm] | tip površi [mm] | udaljenost centara [mm] | razlika radijusa [%] | rastojanje između ravni [mm] | za cilindre [mm] | za ravni [mm] |
| Prirub. 1 | 10^{-17} | 0,00005 | 0,02 | 500 | 3 | 0,3 | 0,3 | 0,4 | 0,3 |
| Prirub. 2 | 10^{-17} | 0,00005 | 0,02 | 100 | 1 | 0,1 | 0,3 | 0,4 | 0,3 |
| Vratilo | 10^{-18} | 0,00005 | 0,02 | 350 | 5 | 0,1 | 0,05 | 0,3 | 0,3 |



Slika 3. Grafički rezultati primene predloženog algoritma nad sintetizovanim oblakom prirubnice 1: a) faza segmentacije; b) faza grupisanja; c) konačni rezultat nakon izdvajanja



Slika 4. Grafički rezultati primene predloženog algoritma nad sintetizovanim oblakom prirubnice 2



Slika 5. Grafički rezultati primene predloženog algoritma nad sintetizovanim oblakom vratila

6. ZAKLJUČAK

U radu je predstavljen metod za prepoznavanje cilindara i ravni u trodimenzionim oblacima tačaka. Metod je razvijen prvenstveno sa potrebom procesiranja nestruktuiranih oblaka tačaka koji reprezentuju određenu grupu mašinskih delova široke upotrebe čija je struktura sastavljena od cilindara i ravni. Predloženi proces prepoznavanja izvodi se procesiranjem oblaka u tri osnovne faze.

Prvu fazu čini algoritam automatske segmentacije oblaka koji prirada grupi algoritama segmentacije sa širenjem regiona počev od jedne izabrane tačke. Za prvu početnu tačku algoritam uzima tačku sa najvećom z koordinatom. Zatim se dodaje jedna po jedna okolna tačka i na osnovu osobina singularnosti informacione matrice sistema i pripadnosti tačaka estimiranoj površini vrši se širenje regiona u dva koraka. Ova faza procesa opisana je u odeljku 2.

Druga faza procesa (odeljak 3) obuhvata gupisanje presegmentiranih oblasti, razvrstavanje površi po tipu i prepoznavanje samih cilindara i ravni (estimacija njihovih parametara). Osnov ove faze čine modifikovani algoritam robusnog prepoznavanja cilindara iz oblaka tačaka kao i razvijeni algoritam precizne estimacije parametara jednačine ravni.

U trećoj fazi, koja je predstavljena u odeljku 4, vrši se izdvajanje cilindara i ravni prepoznatih u prethodnom koraku i formiraju se konačni rezultati. Ova faza unapređuje segmentirane oblasti i dovodi do bolje grafičke interpretacije dobijenih rezultata prepoznavanja.

Celokupan algoritam prepoznavanja cilindara i ravni je eksperimentalno verifikovan procesiranjem tri sintetizovana oblaka tačaka čime je pokazano su polazni kriterijumi za razvoj ovakvog procesa ispunjeni. Dobijeni rezultati mogu se smatrati veoma dobrim i treba naglasiti da je npr. maksimalna greška estimacije radijusa prepoznatih cilindara manja od 0.5%.

Osnovni nedostatak algoritma može se ogledati u broju potrebnih vrednosti pragova. Pregledom tabele 1 može se zaključiti da algoritam poseduje veliki broj (devet) pragova za podešavanje, ali i da su vrednosti nekih od njih jednake i da se one mogu postaviti kao konstante u algoritmu. Takođe, vrlo je izvesno da se eventualno unapređenje prkazanog algoritma može ostvariti postavljanjem određenih zavisnosti nekih pragova i karakteristika oblaka ili nekih pragova međusobno. Na taj način bi se smanjio ukupan broj parametara za podešavanje i povećao nivo autonomnosti samog procesa. Ovo će biti tema budućih istraživanja.

7. LITERATURA

- [1] Varady, T., Martin, R., R., Cox, J.: *Reverse engineering of geometric models – an introduction*, CAD, 29, p.p. 255-268, 1997.
- [2] Rabbani, T., van den Heuvel, F., A., Vosselman, G.: *Segmentation of point clouds using smoothness constraint*, ISPRS Commission V Symposium 'Image Engineering and Vision Metrology', p.p. 248-253, Dresden, Septembar 2006.
- [3] Petitjean, S.: *A Survey of Methods for Recovering Quadrics in Triangle Meshes*, ACM Computing Surveys, 34/2, p.p. 211-262, 2002.
- [4] Hou-Chuan, L., Yi-Hong, C., Jiing-Yih, L.: *Development of feature segmentation algorithms for quadratic surfaces*, Advances in Engineering Software, 40/10, p.p. 1011–1022, 2009.
- [5] Lavoue, G., Dupont, F., Baskurt, A.: *Curvature Tensor Based Triangle Mesh Segmentation with Boundary Rectification*, Proceedings of the Computer Graphics International, p.p. 10-25, Crete, IEEE, 2004.
- [6] Viera, M. Shimada, K.: *Surface mesh segmentation and smooth surface extraction through region growing*, Computer Aided Geometric Design, 22/8, p.p. 771-792, 2005.
- [7] Jakovljevic, Z., Markovic, V.: *Recognition of one class quadric surfaces from unstructured point cloud*, Proceedings of the 8th International Working Conference “Total Quality Management – Advanced and Intelligent Approaches”, p.p. 353-360, Belgrade, JUSK Belgrade, June 2015.
- [8] Jakovljevic, Z., Markovic, V.: *Recognition of quadrics from 3D point clouds generated by scanning of rotational parts*, Journal of Production Engineering, 19/1, p.p. 65-68, 2016.
- [9] Jakovljevic, Z., Markovic, V., Puzovic, R., Majstorovic, V.: *Recognition of one class of quadrics from 3D point cloud*, Procedia CIRP, 57, p.p. 292-297, 2016.
- [10] Ying, X., Yang, L., Kong, J., Hou, Y., Guan, S., Zha, H.: *Direct least square fitting of ellipsoids*, Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), p.p. 3228-3231, Tsukuba, 2012.
- [11] Reza, A., Sengupta, A.S.: *Least square ellipsoid fitting using iterative orthogonal transformations*, Applied Mathematics and Computation, 314/C, p.p. 349-359, 2017.
- [12] Nurunnabi, A., Sadahiro, Y., Lindenbergh, R.: *Robust cylinder fitting in three-dimensional point cloud data*, The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 17/1, p.p. 63-70, 2017.
- [13] Hubert, M., Rousseeuw, P.J., Branden, V.K.: *ROBPCA: A New Approach to Robust Principal Component Analysis*, Technometrics, 47/1, p.p. 64-79, 2005.
- [14] Gander, W., G.H., Golub, Strebel, R.: *Least-Squares Fitting of Circles and Ellipses*, BIT Numerical Mathematics, 34/4, p.p. 558-578, 1994.

Marković, V., Jakovljević, Ž., Budak, I.

RECOGNITION OF CYLINDERS AND PLANES FROM 3D POINT CLOUDS

Abstract: *This paper presents a method for cylinders and planes recognition from unstructured point clouds. Proposed recognition process consists of three stages. The first stage is automatic segmentation based on region growing from one chosen seed point. Region growing conditions are based on scatter matrix properties and belonging of points to the surface whose parameters are estimated using least squares method. The second stage relates to merging of over-segmented regions and recognized surfaces parameters estimation. Utilization of modified robust cylinder fitting algorithm and accurate plane parameters estimation gives good performances to this stage. In the third stage, algorithm processes initial point cloud again, extracts recognized primitives and gives accurate results. The primary goal of the proposed method is recognition of cylinders and planes from point clouds that represent mechanical parts. Accordingly, three synthesized point clouds were employed for algorithm verification.*

Key words: *reverse engineering, point cloud processing, segmentation of surfaces, cylinders recognition, planes recognition*