

SVEUČILIŠTE U ZAGREBU

FAKULTET STROJARSTVA I BRODOGRADNJE

HR-10000 Zagreb, Ivana Lučića 5

(ak. god. 2006/07.)

Seminarski rad iz kolegija:

Struktura i svojstva materijala

KVANTITATIVNA ANALIZA MIKROSTRUKTURE

Mentor:

prof. Zdravko Schauperl

Student:

Tomislav Deković

35981186

5-ps-oskval

Sadržaj

Popis slika.....	2
Popis tablica	3
Uvodni dio.....	4
Razvoj računalne analize mikrostrukture.....	5
Računalna reprezentacija grafičkih podataka	7
Praktični pristup kvantitativnoj analizi.....	12
Tipične korekcije slike kao priprema za analizu.....	13
Dinamičko programiranje – ključ računalne analize slike	16
Dinamičko programiranje – automatsko prepoznavanje oblika	19
Tipični primjeri metalografske analize	21
Mjerenje debljine prevlake.....	21
Mjerenje veličine zrna.....	23
Određivanje udjela faza.....	23
Zaključak.....	25
Popis korištene literature	26

Popis slika

Slika 1 – uvećani detalj slike	8
Slika 2 – vrijednosti svjetline piksela; 8-bitni decimalni zapis	8
Slika 3 – promjene unesene JPEG kompresijom	10
Slika 4 – slika uzorka prije thresholdinga	16
Slika 5 – uzorak nakon thresholdinga	16
Slika 6 – pronalaženje najkraćeg puta pomoću dinamičkog programiranja	17
Slika 7 – grafički prikaz podproblema	18
Slika 8 – definiranje ruba uzorka	19
Slika 9 – početna slika uzorka	21
Slika 10 – thresholding žutom ravninom	22
Slika 11 – dodavanje mreže	22

Popis tablica

Tablica 1 – ovisnost dubine boja i raspoložive game boja	8
Tablica 2 – različiti načini desaturacije	13
Tablica 3 – najčešće korišteni kernel filtri	15

Uvodni dio

U velikom broju praktičnih problema dvodimenzionalna fotografija nekog objekta može se promatrati u formi konkretnog pitanja osobi koja tu fotografiju promatra. Naravno, pitanje ovisi o konkretnoj primjeni i naravi same fotografije ali osnovni principi su za sve praktične primjere analize slike isti. Ako se radi o policijskoj fotografiji pitanje će vjerojatno biti o kojem se tipu i marki automobila radi te koje su oznake na registarskoj pločici, u slučaju uzorka biopsije nekog pacijenta pitanje će biti mogu li se ustanoviti kancerogene promjene na tkivu, i konačno u slučaju fotografije mikrostrukture uzorka pitanje može biti veličina zrna promatranog metala. Svi nabrojeni primjeri (i velika većina ostalih primjera analize slike) imaju dvije važne stvari zajedničke:

- na postavljeno pitanje moguće je potpuno odgovoriti kvantitativno, ili u općem slučaju u obliku rečenice;
- odgovor na postavljeno pitanje se na određeni način nalazi na slici, i postoji konkretni način kako se dolazi do njega.

Ovisno o konkretnoj primjeni i učestalosti analize dolazimo do potrebe da se do odgovora na ova pitanja dođe manje ili više automatski iz više razloga. Prvi razlog je trajanje i smanjivanje napora laboranta pri analizi uzorka. Drugi razlog, koji je najmanje isto toliko važan, je točnost dobivenih rezultata – pokazalo se da je računalna analiza kvantitativnih oblika informacija puno točnija i pouzdanija (manje podložna greškama, ima mnogo veću ponovljivost rezultata itd.).

Da bi računalo moglo obaviti ovakve zadatke umjesto ljudi potrebno je ispuniti neke uvjete da bi problem prilagodili računalnom "načinu razmišljanja". Konkretno, potrebno je razviti određene algoritme po kojima će računalo analizirati zadani objekt (zapravo sliku zadanog objekta). Da bi računalo moglo analizirati svojstva analiziranog objekta po nekom algoritmu najvažnija stvar (i najsloženija kod izrade dekodirajućeg algoritma) je nalaženje odgovarajućeg operatora koji je prikladan za detekciju tražene informacije. Rješenje ovog problema, algoritmi i analitički pristup su glavnina problema kojima se bavi znanstvena disciplina kvantitativne analize slike (QIA – Quantitative Image Analysis).

Od cijelog područja kvantitativne analize slike koje je vrlo široko i uključuje različite primjene (od automatiziranog video nadzora do naprednih medicinskih pretraga poput računalne tomografije) u ovom seminaru težište će biti postavljeno na kvantitativnu analizu mikrostrukture uzoraka u metalografskom laboratoriju. U usporedbi sa klasičnim metodama metalografske analize računalna analiza mikrostrukture donosi brojne prednosti kao što su brzina, jednostavnost, točnost i pouzdanost rezultata, fleksibilnost, veći broj ispitivanja i mogućnost vršenja ispitivanja koja manualnim pristupom nisu bila niti zamisliva.

Razvoj računalne analize mikrostrukture

Nemoguće je razmatrati bilo kakvu analizu mikrostrukture bez osvrta na osnovne principe na kojima se zasniva cijela metalografska analiza [7]. Od prvih istraživanja u 19. stoljeću kad je Anosov prvi put upotrijebio mikroskop da bi analizirao materijal od kojeg je napravljen Damaskov nož (1841. god) i njegova suvremenika Delesse-a u čijim se radovima prvi put pojavljuje tvrdnja o jednakosti volumenskog i površinskog udjela. Njegovu tvrdnju eksperimentalno je dokazao Sorby na način da je projekciju kamenja na papiru izrezao i zatim vaganjem izrezanih dijelova papira dokazao analogiju između volumenskih i površinskih udjela ($A_A = V_V$). Na njihova istraživanja se nastavlja Rosiwal 1898. sa svojom teorijom o analogiji linearnih i volumenskih udjela. Kasniji radovi su ovu povezanost proširili do danas poznatog oblika na kojem se zasniva stereometrija:

$$P_P = L_L = A_A = V_V .$$

Prva istraživanja koja dokazuju povezanost volumnih udjela ugljika u čeliku sa njegovim mehaničkim svojstvima objavio je Albert Sauver 1896. godine. Skoro sto godina nakon prvih metalografskih istraživanja mikrostrukture Thompson (1930.) i Glagolev (1931.) su neovisno jedan o drugome dokazali jednakost između volumnih udjela i udjela točaka ispitne mreže ($P_P = V_V$). Drugim riječima, brojanjem točaka u kojima se određeni elementi podudaraju se ispitnom mrežom možemo odrediti volumenski udio tog elementa u mikrostrukтури. Ova povezanost je s vremenom postala osnova za najznačajnija metalografska mjerenja. To je dugo bio najlakši i najpouzdaniji način da se ustanovi volumni udio pojedinih konstituenata u strukturi materijala.

Ako sad povučemo paralelu na najnoviji razvoj računalne analize (slike) mikrostrukture vidjet ćemo da se ništa revolucionarno nije dogodilo: i dalje udjele faza i konstituenata u materijalu određujemo brojanjem točaka, samo što to više ne radimo manualno nego umjesto nas točke (brže i preciznije) broji specijalizirani softver.

Razvojem televizijske tehnike (koja je nastala na temeljima radarskih sustava nastalih tijekom prvog svjetskog rata), odnosno razvojem vizualizacije slike iscrtavanjem slike pomoću elektronskog snopa otkrile su se nove mogućnosti napretka velikog broja znanstvenih disciplina. Između ostalih i mikroskopije koja je dobila važnu mogućnost da se slika iz mikroskopa u elektroničkom obliku trajno pohrani i prenosi do šireg broja ljudi. Naravno, i prije je bilo moguće fotografirati mikroskopsku sliku ali je za neke primjene taj proces bio prespor (razvijanje filma i slika je predugo trajalo), dok je televizijska tehnika omogućavala trenutno prikazivanje mikroskopske slike širem timu znanstvenika.

Nakon ovih početnih koraka i daljnjeg razvoja televizijske tehnike počeo je skokovit napredak analize mikrostrukture (paralelno sa razvojem kemijskih, medicinskih i sličnih polja mikroskopije). Usporedno s daljnjim razvojem

televizijske tehnike počeo je i razvoj jednog još važnijeg područja: računala. Razvojem računala došlo je do i revolucije u televizijskoj grafici (računala su nudila fleksibilniju podršku za grafički dio televizijske produkcije, titlova, natpisa i sl.).

U laboratorijima Metals Research Ltd. u Cambridgeu 1963. godine razvijen je prvi analitički sustav "Quantiment A" (bio je poznat i pod akronimom QTM - Quantitative Television Microscope), koji je kao ulaznu informaciju koristio sliku preuzetu s televizijske kamere. Sustav je zapravo služio samo kao denzitometar (mjerjenje gustoće tamnih točaka), ali to je bio početak vremena automatizacije. Nasljednik ovog sustava bio je "Quantiment B" koji je ujedno bio i prvi komercijalno uspješan sustav za analizu slike. Quantiment B je imao mogućnost stereoloških mjerenja cijelog vidljivog polja mikroskopske slike. S vremenom je Metals Research postao IMANCO Cambridge Instruments, a nakon toga je promijenio ime u Leica.

Prvi sustav koji je imao mogućnost spremanja cijele crno-bijele mikroskopske slike bio je Bausch and Lomb-ov "QMS" koji je predstavljen 1968. Uz pomoć svjetlosnog markera po prvi put je bilo moguće vršiti konkretna mjerenja izravno na slici uzorka. Nakon toga predstavljen je i prvi potpuno digitalni sustav za analizu slike "Quantiment 720" koji je razvila tvrtka IMANCO (tada već bivši Metals Research Ltd.) 1969. godine.

Razvoj računala omogućio je prilagodbu i druge intervencije nad mikroskopskom slikom koje su dovele do točnijih rezultata i olakšale isticanje važnih informacija u slici mikrostrukture. Grupa pariških znanstvenika u Ecole de Mines je prva počela s razvojem matematičko-morfološke analize i analize tekstura. Osim toga, oni su i prvi razvili metode erozije slike (uklanjanje suvišnih informacija iz slike), softversko uvećanje slike, skeletonizaciju i najznačajnije od svega – korištenje Booleovih binarnih operacija tijekom pripreme slike. Prvi komercijalni sustav koji je koristio sve ove nove mogućnosti bio je Leitz-ov "TAS" (Texture Analysis System) koji je predstavljen 1974. godine.

Nakon ovih početnih koraka istraživanja (i daljnji razvoj računala) rane osamdesete godine prošlog stoljeća su donijele softverski bazirane sustave koji su se sve manje oslanjali na specijalizirani hardver. Sadašnji trend je okrenuo stvari obrnuto – sustavi za analizu slike se opet počinju oslanjati na specijalizirani hardver (ovoga puta mnogo fleksibilniji) koji omogućuje prilagodbe slike i isticanje važnih informacija u realnom vremenu.

Računalna reprezentacija grafičkih podataka

Kako bismo mogli razmatrati algoritme prema kojima računalo analizira mikroskopsku sliku potrebno je prethodno razjasniti barem osnovne pojmove računalne grafike. Pri tome je važno spustiti se na najnižu moguću razinu i prisjetiti se von Neumann-ovog modela računala. Bez suvišnih detalja, bitno je samo naglasiti da je računalo čisto matematički stroj koji u stvari provodi samo osnovne matematičke operacije zbrajanja i množenja (koje se opet svodi na zbrajanje broja sa samim sobom onoliko puta koliki je množitelj). Princip rada von Neumann-ovog stroja se može svesti u jednu rečenicu: podatak spremljen na adresi A pribroji podatku na adresi B i njihovu sumu spremi na adresu C. Naravno, von Neumann je svoj model računala predstavio u vrijeme kad nije bilo tranzistora niti elektronskih cijevi – njegovo računalo je bilo mehaničko. Napredak elektronike je omogućio da se procesoru računala zadaju instrukcije više razine od samog pribrajanja, ali osnovni princip se i dalje zadržao. Dakle, sve operacije koje računala danas rade mogu se svesti na puko zbrajanje brojeva (po određenom redosljedu i jako brzo, ali ipak samo na zbrajanje). Razlika u današnjim procesorima je ta što im se naredbe zadaju na višoj logičkoj razini (asemblerski programski jezici), a oni sami pomoću interne logike asemblerske naredbe razlažu na slijed osnovnih instrukcija.

Iz ovog razmatranja slijedi da su svi podaci koji se nalaze u računalu kvantificirani i reprezentirani brojevima (konkretno brojevima u binarnom zapisu). Bilo koja vrsta podataka, pa tako i slike nalaze u računalnoj memoriji su kvantificirani u obliku brojeva i sve operacije nad tim podacima su matematički modelirane.

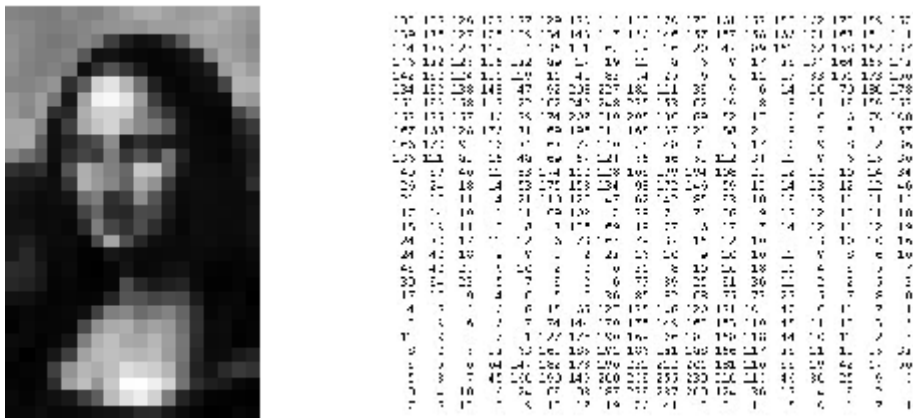
Računalna grafika (slika) se zapravo pojavljuje u dva principijelno različita oblika: vektorska i bitmap grafika. Vektorska grafika se koristi za interpretaciju pravilnih geometrijskih likova i ima svoje velike prednosti u području u kojem se koristi, ali za područje kojim se bavi ovaj rad nema nikakav značaj. Ulazni podaci, odnosno mikroskopska slika uzorka koju treba analizirati, uvijek se nalazi u bitmap obliku. Bitmap slika je skup podataka strukturiran na način da je svaki element (točka, piksel) reprezentiran brojčanom vrijednošću.

Elementi slike (pikseli) su poredani u redove i stupce, a kod većine bitmap formata navedeni su red po red s lijeva na desno (kod nekih formata moguće je i drukčije definiranje podataka o pojedinom pikselu, npr. BITMAP¹ formata koristi se još "interlaced" način navođenja redova kod kojeg su najprije navedeni neparni pa zatim parni redovi). O broju piksela po pojedinom retku i stupcu ovisi rezolucija slike (nije povezana sa optičkom rezolucijom leće) koja definira količinu informacija koju je moguće pohraniti u slici. Slika veće rezolucije može pohraniti

¹ BITMAP – standardni format slika u Windows sustavima; najčešće bez kompresije (u nekim slučajevima podržava RLE kompresiju); podržava najviše 32-bitnu dubinu boja

više informacija o detaljima nego slika niže rezolucije. Npr. Slika 1 prikazuje detalj poznate slike Mona Lisa, a vidljivi kvadratići (pikseliziranost slike) su posljedica male rezolucije slike).

Slika 1 [11] prikazuje uvećani detalj poznate slike, dok Slika 2 prikazuje kvantificirane vrijednosti svjetline pojedinih piksela iste slike. Zapravo, u računalnoj memoriji nisu pohranjeni decimalni brojevi nego binarni (konkretno binarna reprezentacija heksadecimalnih brojeva). Pri manipulaciji slikama se skoro redovito koristi heksadecimalni brojevni sustav koji je najpogodniji za operacije nad slikama (traži manje procesorskog rada i zauzima manje memorije u odnosu na decimalni zapis). Iako je zbog velikog dinamičkog raspona i decimalnog zapisa 8-bitne dubine boja moguće vidjeti povezanost piksela slike i iznosa svjetline pojedinog piksela nemoguće je očekivati da ljudsko oko raspozna o čemu se radi na slici, a možda ovaj primjer nudi i odgovor na pitanje zašto računala ne znaju za umjetnost.



Slika 1 – uvećani detalj slike Slika 2 – vrijednosti svjetline piksela; 8-bitni decimalni zapis

Kakva će biti brojčana vrijednost pojedinog piksela slike ovisi o više parametara. Najvažniji parametar je svakako dubina boja koja je presudan faktor koji utječe na količinu informacija sadržanih u slici, odnosno njezinom najmanjem elementu – pikselu. Za duotonsku sliku dovoljan je jedan bit po pikselu (radi se o slici čije točke mogu biti samo crne ili bijele, nema nijansiranja). Crno-bijela slika već traži 8 bita itd.

BIT	DUBINA BOJA (BIT DEPTH)
2 ¹	crno ili bijelo
2 ⁸	256 nijansi sive
2 ²⁴	16.7 milijuna boja
2 ³²	4.29 milijuna boja

Tablica 1 – ovisnost dubine boja i raspoložive game boja

Tablica 1 prikazuje ovisnost dubine boja i broja raspoloživih boja odnosno nijansi koje je moguće prikazati. Iz gornje tablice vidimo da je količina informacija koje nosi jedan piksel izravno povezana sa veličinom datoteke koja sadrži sliku. Drugi parametar važan za veličinu datoteke je sam format datoteke, koji određuje način na koji će informacije o slici biti procesirane i zapisane u datoteku. Principijelno, razlikuju se formati sa gubitkom informacija ili bez gubitka. Odgovarajući format ovisi o namjeni slike, za veliku većinu namjena nisu presudne apsolutno sve informacije, pa je moguće primjenom određenih algoritama (koji su razvijeni u skladu sa istraživanjima percepcije ljudskog oka – utvrđeno je da ljudsko oko slabije razlikuje neke nijanse i da bolje razlikuje male promjene svjetline nego male promjene nijanse boje i sl.) odbaciti dio informacija, a da to ne utječe puno na način na koji ljudsko oko vidi sliku. Najrašireniji takav format sa gubitkom kvalitete je JPEG², koji pruža mogućnost smanjivanja originalne datoteke na samo 20-30 % originalne veličine bez vidljivih razlika na samoj slici.

U našoj primjeni moguće je uz određene uvjete (smanjeni zahtjevi na preciznost, ograničena kompresija itd.) koristiti JPEG format za samu analizu slike, ali preporuča se korištenje nekog formata koji ne odbacuje informacije prikupljene senzorom digitalne kamere (TIFF³, RAW⁴ i slični). JPEG format se često koristi za spremanje slika koje će biti tiskane u izvještaju, ili za spremanje slika međustupnjeva pripreme slike za analizu da bi mogli istaknuti neku zanimljivu pojavu. Ipak, za važna mjerenja i analize potrebno je obratiti pažnju na format slika i na eventualni gubitak informacija i sustavne pogreške koje taj gubitak može unijeti u rezultate.

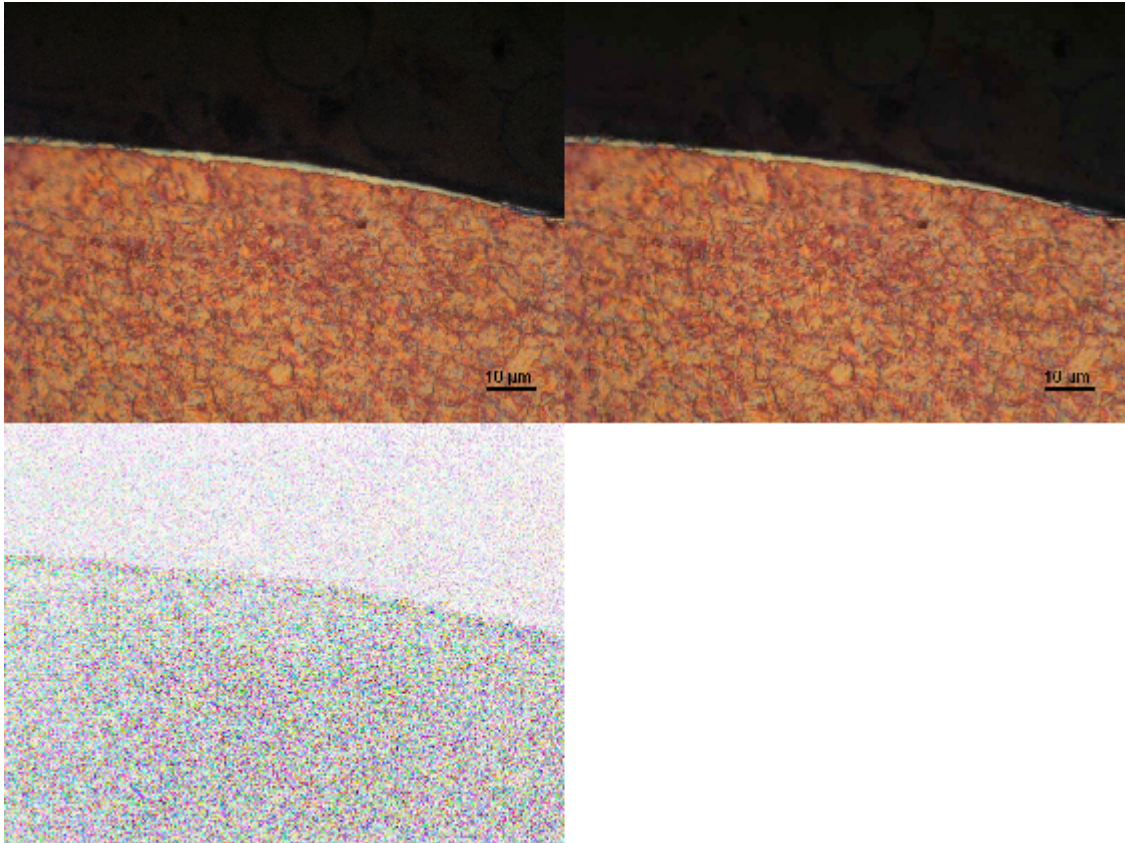
Slika 3 prikazuje razlike koje unosi JPEG kompresija u sliku mikrostrukture uzorka (snimljeno u sklopu vježbi iz kolegija Laboratorijsko ispitivanje materijala, FSB 2006). Detalj gore lijevo predstavlja originalnu JPEG sliku snimljenu pod svjetlosnim mikroskopom (povećanje 1000x) digitalnom kamerom Olympus C5050Z (stupanj kompresije kamere je postavljen na "Super", koji odgovara JPEG kvaliteti 90), dok desni detalj prikazuje istu tu sliku na kojoj je primijenjen viši stupanj JPEG kompresije (JPEG kvaliteta 20). Originalna datoteka je približno 7 puta veća od naknadno komprimirane uz isti broj piksela, a razlike je golim okom teško raspoznati. Da bi bilo lakše vidjeti razliku ispod ovih slika je prikazana grafika koja prikazuje razlike između njih, pri čemu potpuno bijela područja predstavljaju nepromijenjene dijelove a tamnija područja prikazuju promjene (što je određeno područje tamnije to je veća promjena unesena kompresijom). Ovo nas dovodi od zaključka da, iako subjektivno gledano slika

² Joint Photographic Experts Group format – najčešći format slika za opću upotrebu, koristi dobar i fleksibilan algoritam kompresije i omogućava odličan omjer veličine datoteke i kvalitete prikaza; vrlo široko podržan od svih vrsta uređaja

³ Tagged-Image File Format – drugi najčešći format za razmjenu slika, ima mogućnost korištenja nekoliko vrsta lossy i lossless kompresije, podržava transparentnost i slojeve (layer) slike; izvorno zamišljen kao univerzalni format razmjene slika između različitih aplikacija i operativnih sustava

⁴ RAW – format slika koji bilježi originalnu informaciju detektiranu senzorom digitalnog fotoaparata (bez automatskih korekcija pri snimanju), vrlo prikladan za situacije kad nam je potrebna što preciznija reprodukcija slike

nije bitno promijenjena, računalna analiza pokazuje znatne razlike koje konkretno za svrhu metalografske analize ne bismo smjeli zanemariti.



Slika 3 – promjene unesene JPEG kompresijom

Kad se radi o crno bijeloj slici tada se vrijednosti piksela izražavaju samo svjetlinom koja u potpunosti definira odgovarajuću nijansu sive, ali kod slika u boji moramo definirati više parametara za svaki piksel. Reprodukcijska slika u boji se principijelno može podijeliti na subtraktivni i aditivni princip reprodukcije. Ova dva principa se razlikuju prema izvoru svjetlosti koju oko detektira gledajući u sliku. Razliku je najlakše prikazati kroz dva tipična praktična primjera.

Subtraktivni princip je najlakše objasniti kroz promatranje isprintane slike. Svjetlost (tzv. bijela svjetlost) pada na sliku iz vanjskog izvora (tipično dnevna svjetlost ili neka lampa) i ovisno o boji određenog dijela slike dio spektra bijele svjetlosti se odbija do oka promatrača. Npr. crveno područje na slici upija sve dijelove spektra osim crvenog i reflektira crvenu svjetlost u oko promatrača. Taj princip upijanja određenih dijelova spektra se naziva subtraktivni (oduzimajući).

Aditivni (zbrajajući) princip je upravo suprotan prethodnome. Tipični primjer je prikaz boja na zaslonu računala ili televizora. Ovdje se radi o tome da se emitiraju najčešće tri boje (tri dijela spektra) u točno određenim omjerima za svaku nijansu neke boje. Tradicionalno se radi o crvenoj, zelenoj i plavoj boji koje

su u upotrebi od prvih televizora u boji. Ti dijelovi spektra se zbrajaju i kod oka promatrača stvaraju dojam jedne određene boje.

Ovakvo definiranje boja je preneseno i na područje računala, pa je jedan od važnijih načina da se definira određena boja baziran na tzv. RGB modelu. Prema RGB modelu dovoljno je definirati iznos svake pojedine komponente da bi se jednoznačno definirala boja. Često se za izražavanje RGB modela koristi i tzv. HEX (heksadecimalni zapis), kod kojega se umjesto troznamenkastih decimalnih vrijednosti pojedinog kanala koristi dvoznamenkasti heksadecimalni koji zauzima manje memorije i lakše ga je procesirati (umjesto formata RRRGGGBBB koristi se heksadecimalno RRGGBB – decimalni raspon 0-125 odgovara heksadecimalnom 0-FF).

Drugi model definiranja boja je HSV (Hue, Saturation, Value) i zasniva se na definiranju tona boje (hue) koji korespondira sa valnom duljinom svjetlosti, zasićenosti boje (saturation) i svjetline (value) koja je analogna svjetlini crno-bijele slike. Zapravo se najčešći način pretvaranja slike u boji u crno-bijelu provodi baš tako da se zanemare podaci o tonu i zasićenosti uz prenošenje vrijednosti svjetline.

Praktični pristup kvantitativnoj analizi

U metalografskoj praksi cijeli posao počinje izdvajanjem i pripremom uzorka. O tim područjima nema smisla puno pisati jer bi za svako od njih opseg ovog seminara bio premali. Ali važno je naglasiti da je priprema uzorka kritična za cijeli proces analize jer ona izravno utječe na krajnji rezultat. Poznato je da je bilo slučajeva kad su laboratoriji povjerovali marketinškim trikovima da će računalni sustavi za analizu slike ispraviti nedostatke uzrokovane lošom pripremom uzorka, pa su na kraju bili razočarani rezultatima čija je ponovljivost pokazala da su nepouzdana. Istina je da napredni algoritmi pripreme i prilagodbe slike mogu ispraviti neke nedostatke, pa čak i male probleme sa dubinskom oštrinom ili nekonzistentnim nagrizanjem po cijeloj površini uzorka, ali veće nedostatke nije moguće ispraviti bez utjecanja na krajnji rezultat koji bi morao biti nepristran i bez vanjskih utjecaja.

Nakon pripreme i nagrizanja uzorka slijedi postavljanje pod mikroskop i snimanje slika. Obično se unaprijed snimi više slika, ovisno o cilju analize određuju se povećanja i detalji koje treba prikazati. Tu je važno napomenuti jedan detalj iz prakse koji će osigurati optimalno korištenje opreme bez suvišnog procesiranja i gubitka vremena, a da laborant bude siguran da ima reprezentativne podatke.

Ovisno o značajki materijala koja će se mjeriti potrebno je prilagoditi povećanje (zajedničko povećanje mikroskopa i digitalne kamere) na način da nam ta određena značajka bude zastupljena sa najmanje 100 piksela na fotografiji. Veće povećanje će značiti i više detalja, ali će povećati sami broj piksela koje je potrebno procesirati u sklopu analize. Npr. za mjerenje veličine zrna čelika potrebno je prilagoditi povećanje na način da nam većina zrna bude zastupljena sa 100 piksela. To će olakšati kasniju prilagodbu slike i omogućiti isticanje važnih područja (lakše odvajanje pojedinih zrna, preciznije prebrajanje i mjerenje njihove veličine).

Kada imamo sliku (fotografiju mikrostrukture) željenog objekta moramo je pripremiti za analizu. Tipični proces računalne analize slike teče po manje-više uvijek istom redoslijedu [6]:

- korekcija slike – balans boja, kontrast, oštrina, desaturacija...;
- stvaranje binarne slike – diskriminacija, thresholding;
- morfološka obrada binarne slike – segmentacija faza, odvajanje pojedinih zrna, označavanje bitnih dijelova...;
- vršenje automatiziranih mjerenja – udio faza, veličina zrna, kružnost odnosno ujednačenost oblika zrna...;
- vršenje interaktivnih mjerenja – mjerenja kod kojih laborant zadaje parametre, odnosno označava položaje mjernih točaka i vrste mjerenja;
- obrada i prezentacija podataka – najčešće u vidu laboratorijskog izvještaja.

Da bi došli do konkretnih kvantitativnih podataka sadržanih u slici objekta potrebno je iz te slike eliminirati nepotrebne informacije i istaknuti informacije koje sadrže tražene podatke. Ovo je proces koji se obavlja u više stupnjeva kako bi bio što točniji i fleksibilniji. Zahtjev za točnošću u znanstvenim aplikacijama je sam po sebi razumljiv, a fleksibilnost je potrebna da bi jednom razvijeni algoritam koji se pokaže dobrim u primjeni mogli koristiti za što više različitih primjena.

Najčešće kao ulaznu informaciju imamo sliku s digitalnog fotoaparata. Slike su redovito u boji sa relativno velikom rezolucijom (tipično 4-16 MPix), i u sebi nose veliki broj detalja (informacija) od kojih nisu nužno svi potrebni. Prva operacija nad slikom je biranje područja interesa, pri kojem se izbacuju dijelovi slike koji nisu važni za analizu ili bi mogli unijeti grešku u rezultate.

Dio slike koji ćemo analizirati se izrezuje (eng. cropping), a zatim se nad njim primjenjuju korekcije koje će istaknuti značajna svojstva važna za analizu. Za tu svrhu ćemo proučiti neke tipične operacije i osnovne principe na kojima se zasnivaju.

Tipične korekcije slike kao priprema za analizu

Desaturacija

Jedna od najvažnijih operacija koja se primjenjuje u više od 90% sustava za računalnu analizu slike je desaturacija, odnosno pretvaranje slike u boji u crno-bijelu sliku (nijanse sive). Time se eliminiraju nepotrebne vizualne informacije jer je osnova analize u principu geometrijska, a ne vizualna.

Desaturacija se može provesti na više načina, a najjednostavniji je od njih je jednostavno zanemarivanje informacija o tonu i zasićenosti boje HSV modela (klasična desaturacija) pri čemu se vrijednost svjetline piksela prenosi nepromijenjena u crno-bijelu sliku. Često se za desaturaciju koristi samo jedan od RGB kanala na način da se odabere crveni, zeleni ili plavi kanal koji najbolje predstavlja traženu morfologiju a ostali kanali se zanemaruju. Osim ova dva načina razvijeno je još mnogo drugih, ali oni se zapravo svode na jedan od osnovnih načina uz prethodno provođenje nekih drugih filtara. Osnovne načine desaturacije prikazuje Tablica 2.

BOJA		CRNO-BIJELI REZULTAT OPERACIJE			
model zapisa	osnovna boja	klasična desaturacija	preko crvenog kanala	preko zelenog kanala	preko plavog kanala
RGB	197 124 69	197	197	124	69
HEX RGB	C5 7C 45	C5	C5	7C	45
HSV	26 65 77	0 0 77	0 0 77	0 0 49	0 0 27

Tablica 2 – različiti načini desaturacije

Ovisno o pristupu, moguće je desaturaciju provesti kao prvu operaciju nad slikom (i to se često primjenjuje) ili nakon ostalih operacija korekcije. Najčešće se desaturacija primjenjuje nakon podešavanja kontrasta.

Konvolucijski filtri (kernel filtri)

Pod nazivom konvolucijski filtri podrazumijevamo filtre koji pri obradi slike novu vrijednost piksela izračunavaju u ovisnosti o staroj vrijednosti i vrijednostima okolnih piksela $X' = f(X, Y_{ij})$. Logika iza ovoga je prilično jednostavna i zasniva se na tome da bilo koja smisljena operacija koju želimo provesti nad dijelom slike (pikselu) ovisi o okolnim dijelovima slike (okolnim pikselima). Da bi objasnili princip djelovanja, krenut ćemo od jednostavne obrade crno-bijele slike. U najjednostavnijem slučaju uzimaju se u obzir samo pikseli iz neposredne okoline, traženog piksela. Uzmimo kao primjer segment slike koji na desnom rubu ima prijelaz na tamno područje, ako na taj segment primijenimo filter:

$$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \times \begin{bmatrix} 240 & 110 & 19 \\ 255 & 120 & 20 \\ 250 & 130 & 21 \end{bmatrix} = \begin{bmatrix} 240a & 110b & 19c \\ 255d & 120e & 20f \\ 250g & 130h & 21i \end{bmatrix}$$

Matrica sa elementima a, b, c,... naziva se kernel, i o njezinim elementima ovisi kakva će biti promjena slike, odnosno dijela slike. Vrijednosti elemenata za neke tipične crno-bijele kernel filtre su (navedeni su engleski nazivi zbog jasnoće):

operator	kernel	napomena
LOW PASS MEAN	$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} * \frac{1}{9}$	SMOOTHING zaglađivanje
MEDIAN	$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$	SMOOTHING zaglađivanje (e se zamjenjuje medijanom vrijednosti a, b, ...i)
GAUSSIAN	$\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$	SMOOTHING zaglađivanje
LAPLACIAN	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	EDGE ENHANCMENT isticanje rubova (prijelaza)

GRADIENT	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	EDGE ENHANCMENT isticanje rubova (prijelaza)
SMOOTH	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$	NOISE REDUCTION smanjivanje šumova

Tablica 3 – najčešće korišteni kernel filtri [7]

Tablica 3 prikazuje najčešće korištene kernel filtre koji se koriste za operacije nad crno-bijelim slikama. U počecima razvoja automatske analize slike ovi su filtri bili usko grlo zbog procesorskog vremena koje zauzimaju, ali s razvojem bržih i matematički moćnijih procesora oni više nisu tako ograničavajući faktor (u usporedbi sa vremenom pripreme i snimanja pod mikroskopom, te izrade laboratorijskog izvještaja vrijeme automatizirane računalne analize je zanemarivo malo).

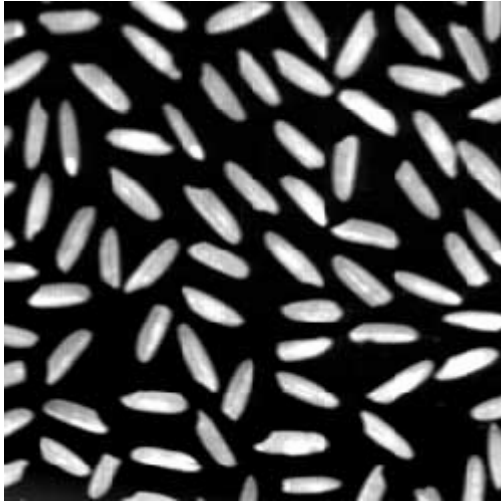
Osim što postoji veliki broj modifikacija gore navedenih filtara, koriste se još i višedubinski filtri (matrice 4×4 i veće) sa različitim utjecajima pojedinih dubina okolnih piksela. Daljnje proširenje je filtriranje slika u boji koje zahtijevaju višedimenzionalni pristup koji također može dati različite rezultate ovisno o stupnju povezanosti pojedinih kanala boja.

Na sličnom principu rade i tzv. delineacijski filtri koji detektiraju prijelaze između svijetlih/tamnih područja ili prijelaze između različitih boja, a može im se i zadati prag diskriminacije (parametar koji kaže koliki iznos promjene će se detektirati kao prijelaz). Ipak, delineacijski filtri nisu dovoljno precizni i pouzdani kad se radi o visokim zahtjevima na preciznost i pouzdanost rezultata. Zbog te činjenice razvijeni su posebni algoritmi koji bolje definiraju položaj različitih pojava u mikrostrukтури. Ti algoritmi su vrlo fleksibilni i zasnivaju se na posebnom pristupu programiranja – dinamičkom programiranju.

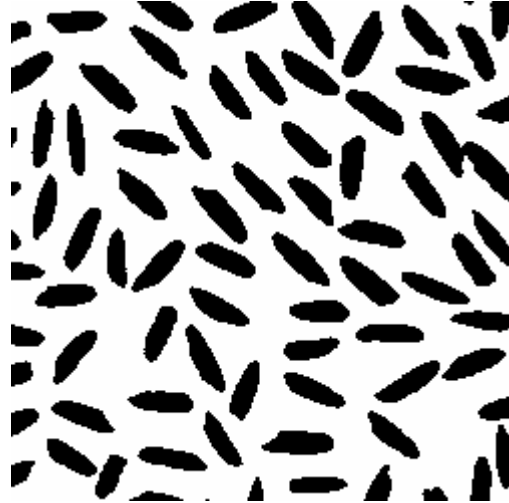
Thresholding – prag diskriminacije

Najvažnija i najkorištenija operacija nad slikama prije same analize je sigurno diskriminacija (eng. thresholding). U postupku eliminiranja suvišnih informacija iz analizirane slike thresholding je najradikalniji, ali i ključni korak. Sve pripremne operacije nad slikom se provode na način da se thresholding filtru što bolje prilagodi sadržaj slike, tj. da se bitni dijelovi slike dovedu do izražaja kako bi ih lakše filtrirali. Thresholding se može provoditi i na slici u boji, ali najčešće se izvodi na crno-bijelim slikama. Ovaj filter je najradikalniji (i samim time kritičan za konačni rezultat) i najznačajniji u postupku pripreme slike za analizu. Radi se o tome da se crno-bijela (grayscale, tonovi sive boje – 8-bitna dubina boja) reducira na binarnu sliku sa crnim ili bijelim pikselima (black and white – 1-bitna dubina boja). Ovaj proces omogućava algoritmima računalne analize da lakše

prepoznaju značajke mikrostrukture, te da ih interno adresiraju kako bi ih mogli analizirati i mjeriti.



Slika 4 – slika uzorka prije thresholdinga [12]



Slika 5 – uzorak nakon thresholdinga [12]

Slika 4 prikazuje uzorak prije thresholdinga (konkretno, na ovoj slici se ne radi o mikrostrukтури materijala nego o riži na tamnoj podlozi) na kojem je vidljivo da bi bilo teško točno definirati granice zrna jer se radi o gradijentnim prijelazima umjesto oštrog prijelaza kojeg će softver prepoznati. Rezultat operacije thresholdinga prikazuje Slika 5, koja je vidljivo bolja baza za bilo kakvu analizu jer ima jasno izražene granice zrna i puno bolji (zapravo idealni) kontrast. Kada kao polazište za analizu imamo ovako pripremljenu sliku lakše je projektirati algoritme pomoću kojih ćemo analizirati mikrostrukтуру.

Dinamičko programiranje – ključ računalne analize slike

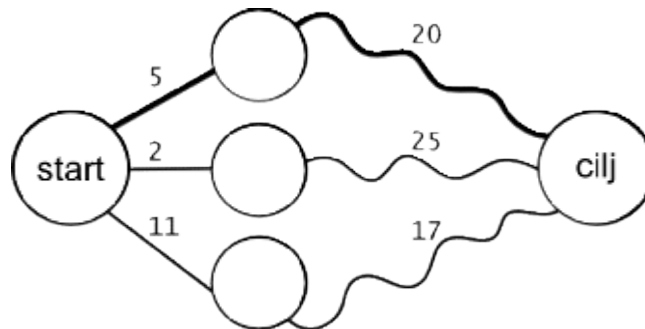
Pri razvijanju algoritama analize slike koristi se poseban pristup rješavanja problema – dinamičko programiranje. Sam pojam programiranja ovdje se ne koristi striktno u smislu računalnog programiranja nego više u smislu pristupa rješavanja skupa konkretnih međusobno povezanih problema. Pojam programiranja se ovdje više odnosi na pronalaženje konkretnog plana akcije za rješavanje problema.

Dinamičko programiranje je brz i elegantan pristup rješavanju velikog broja problema koji se tiču nalaženja optimalnog rješenja i općenito rješavanju problema koji se sastoje od većeg broja manjih problema, tzv. "podproblema". Između ostalog koristi se i za rješavanje raznih problema optimiranja, od najjednostavnijeg (nalaženja najefikasnijeg ili najjeftinijeg puta) do najsloženijih (čak se i najsloženiji programi računalnog šaha baziraju na dinamičkom programiranju). Fleksibilnost i efikasnost ovog pristupa je razlog čestom korištenju i razvijenosti teorije dinamičkog programiranja. Pristup problemu dinamičkog programiranja se sastoji od matematičkog i programerskog dijela, ali

ovdje ćemo obraditi samo osnove kako bi pojasnili princip rada računalne analize.

Sam termin dinamičkog programiranja je prvi koristio Richard Bellman 40-ih godina prošlog stoljeća da bi opisao proces rješavanja problema kod kojih je potrebno pronaći veliki broj najpovoljnijih rješenja koja su na neki način međusobno povezana. Tijekom svojih istraživanja cijelu teoriju je unaprijedio i 1953. god. je objavio rad u kojemu je termin dinamičkog programiranja poprimio današnje moderno značenje.

Praktičnost i fleksibilnost dinamičkog programiranja ćemo pokazati na jednostavnim primjerima. Jedan od najčešćih problema je svakako problem pronalazjenja najkraćeg puta. Slika 6 prikazuje samo tri moguća puta da bi bilo lakše razumjeti način rada, dok se u praksi pojavljuju vrlo složeni primjeri sa velikim brojem kombinacija koje treba analizirati. Između točaka su nacrtani putovi sa naznačenom duljinom svakog puta, a zadatak je pronaći najkraći put (u primjeru je označen debljom linijom od dva alternativna puta).

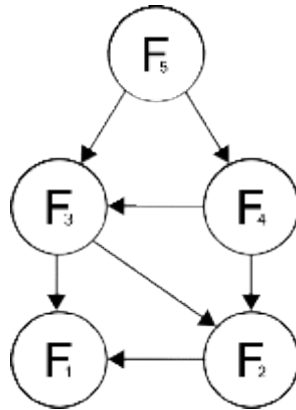


Slika 6 – pronalazjenje najkraćeg puta pomoću dinamičkog programiranja [4]

Princip optimalne podstrukture znači da ćemo optimalna rješenja podproblema koristiti da bi odredili optimalno rješenje cjelokupnog problema. U gornjem primjeru moguće je naći najkraći put na način da najprije odredimo koji je najkraći put do cilja iz svih susjednih točaka, a zatim to koristimo da odaberemo ukupno najkraći put. U općem slučaju problem pronalazjenja optimalne podstrukture koristeći proceduru u tri koraka:

- razbijanje problema u manje podprobleme;
- nalaženje optimalnih rješenja podproblema rekursivnim računanjem;
- korištenje optimalnih rješenja podproblema da bi se odredilo optimalno rješenje osnovnog problema.

Rješenja podproblema se pronalaze na način da se i oni razbijaju na podpodprobleme opet koristeći gornju proceduru, i tako dalje dok se na krajnjoj razini ne dođe do problema kojeg je moguće jednostavno riješiti.



Slika 7 – grafički prikaz podproblema [4]

Ako pretpostavimo da se problem sastoji od preklapajućih podproblema onda je moguće pomoću rješenja tih podproblema riješiti problem više razine. Slika 7 ovo ilustrira grafički: $F_3 = F_1 + F_2$ i $F_4 = F_2 + F_3$ – računanje bilo kojeg od ova dva broja podrazumijeva računanje broja F_2 . S obzirom na to da će za računanje broja F_5 biti potrebno izračunati F_3 i F_4 neefikasan pristup rješavanju bi doveo do toga da se broj F_2 izračunava dva ili više puta, ovisno o broju razina problema. Ovo se događa kod rješavanja svih problema koji se sastoje od preklapajućih podproblema: neefikasan ili neoprezan pristup bi doveo do velikog rasipanja procesorskog vremena za višestruko rješavanje već riješenog problema.

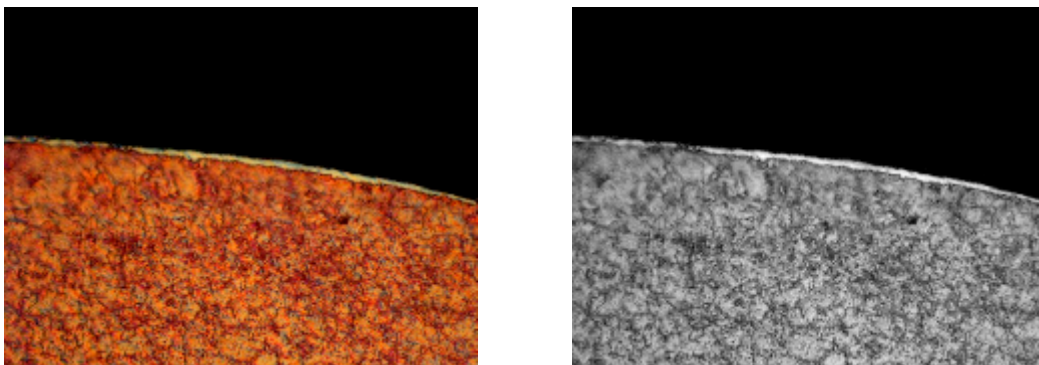
Da bi se ovo izbjeglo rješenja podproblema se pohranjuju, i kad ih ponovo zatrebamo samo ih pozovemo iz memorije. Ovaj pristup se naziva memoizacija (ne memorizacija, iako je i ovaj termin u neku ruku prikladan). Ako znamo da nam neko od rješenja niže razine više neće biti potrebno lako ga je izbrisati iz memorije da bi izbjegli nepotrebno trošenje memorije. U nekim slučajevima možemo čak i unaprijed izračunati rješenja svih podproblema za koje znamo da će nam biti potrebna.

Znači, dinamičko programiranje se za rješavanje problema koristi principima preklapajućih podproblema, optimalne podstrukture i memoizacije međurezultata. Moguće je koristiti dva različita pristupa rješavanju problema pomoću dinamičkog programiranja:

- TOP-DOWN pristup – problem se razbija na podprobleme, ti podproblemi se rješavaju i njihova rješenja se zabilježe za slučaj da ih ponovo zatrebamo. Tu se radi o kombiniranju memoizacije i rekurzije.
- BOTTOM-UP pristup . svi podproblemi koje će biti potrebno riješiti rješavaju se unaprijed, a onda se rješenja kombiniraju da bi se odredilo rješenje problema više razine dok se ne dođe do konačnog rješenja. Ovakav pristup se pokazao boljim u smislu manjeg zauzimanja memorije i manjeg broja pozivanja funkcija, ali u nekim slučajevima je teško unaprijed definirati koje sve podprobleme treba riješiti da bi se došlo do konačnog rješenja.

Dinamičko programiranje – automatsko prepoznavanje oblika

Da bi prethodno objašnjenu teoriju lakše prepoznali koristit ćemo jedan praktični primjer. Slika 8 prikazuje fotografiju mikrostrukture uzorka, i mi želimo pomoću dinamičkog programiranja definirati granicu uzorka. Vidljivo je da je granica uzorka koincidira sa slojem svjetlijeg metala na vanjskom rubu uzorka. Taj svjetliji sloj (svijetla pruga) će nam poslužiti kao nit vodilja: postaviti ćemo problem na način da tražimo putanju s lijevog ruba slike na desno pri čemu će nam zadatak biti da se tijekom putanje što manje mijenja svjetlina slike (to nam osigurava da će putanja pratiti granicu uzorka jer se i iznad i ispod granice nalaze tamnija područja).



Slika 8 – definiranje ruba uzorka

Dakle, problem ćemo riješiti postupno dinamičkim programiranjem u 4 koraka:

1. inicijalizacija f : definiramo početnu točku putanje $(1,197)$ i njezinu svjetlinu (128) ;
2. za svaki stupac j , odnosno redak i na slici koristimo dinamičko programiranje da minimiziramo $C(f)$ uz već prethodno poznatu svjetlinu f iz prethodnog koraka:

$$C(f) = \sum_i \sum_j \left[(Y_{ij} - f_{ij})^2 + \lambda \left\{ \phi(f_{ij} - f_{i-1,j}) + \phi(f_{ij} - f_{i,j-1}) + \phi(f_{ij} - f_{i+1,j+1}) \right\} \right];$$

3. bilježimo položaj svakog rezultirajućeg piksela;
4. ponavljamo korake 2 i 3 do desnog ruba slike, a zatim rekonstruiramo cijelu putanju.

Na ovaj način je moguće uz relativno male modifikacije rješavati vrlo različite praktične probleme. Naravno, ovo je samo pojednostavljenje kako bi lakše shvatili princip djelovanja algoritama za prepoznavanje određenih svojstava analiziranog objekta. Za različite probleme postoje razvijeni posebni algoritmi koji su napredniji od našeg primjera u smislu efikasnosti ili zahtjeva za alokacijom memorije, pa čak i na način da primaju više ulaznih parametara. U gornjem primjeru smo zadali samo jedan parametar – koordinate početne točke. Moguće je napraviti algoritam kojem ćemo zadati još neke dodatne parametre kao što su gornja i donja granica putanje (ili čak pravokutno ili elipsoidno područje), gornja i

donja granica svjetline koju može prihvatiti kao rješenje međukoraka itd. Često se za ovakve probleme koriste naprednije metode kao što su Gibbs sampling, Forward-Backward sampling, simulated annealing [10].

Softverski paketi specijalizirani za analizu slike u sebi sadrže veliki izbor alata za označavanje geometrije objekata koje želimo analizirati, a koji su uglavnom zasnovani na ovakvim algoritmima. Ti alati su prilagođeni specifičnim zadaćama te su vrlo fleksibilni i precizni, ali princip djelovanja im je uglavnom isti ili vrlo sličan našem pojednostavljenom primjeru i svodi se na razbijanje problema na manje podprobleme, njihovo rekurzivno rješavanje i zatim određivanje glavnog rješenja na osnovi rješenja problema niže razine.

Tipični primjeri metalografske analize

Nakon što smo razjasnili principe pomoću kojih računalo obrađuje i priprema sliku za kvantitativnu analizu, te princip pomoću kojeg se utvrđuju položaj, veličina i oblik objekata na slici lakše je razumjeti samu analizu slike. Uz ovo razumijevanje lakše je i efikasno koristiti softver za analizu jer je jedino na taj način moguće intuitivno rješavati teškoće ili vršiti rijetka mjerenja koja nisu automatizirana ili u dovoljnoj mjeri dokumentirana.

Naravno da nema smisla razvijati i teoriju o samim matematičko-statističkim temeljima na kojima se zasniva mjerenje npr. veličine zrna ili debljine sloja prevlake na uzorku jer to već prelazi u neko drugo područje znanosti i bespotrebno bi povećalo opseg ovog rada. S druge strane, temelji i izvod osnovne jednadžbe stereometrije ($P_P = L_L = A_A = V_V$) također je dobro poznat svakome tko je čak i površno upoznat sa metalografijom pa nema potrebe ulaziti u detalje. Isto tako bi bilo promašeno u ovakvom seminaru ulaziti u razloge povezanosti mikrostrukture materijala i njegovih mehaničkih svojstava.

Uz opis temeljnih principa softvera za automatsku analizu slike u slijedećem poglavlju ćemo ukratko opisati neke tipične primjere iz metalografske analize.

Mjerenje debljine prevlake

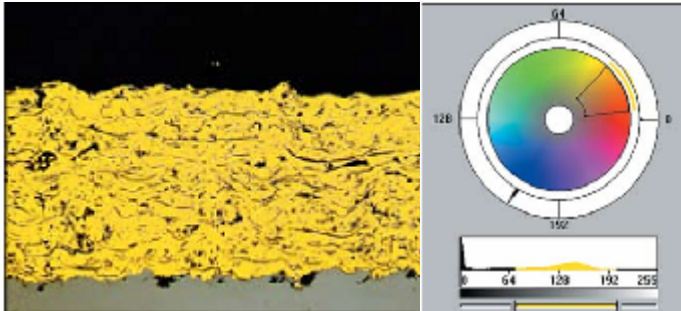
Počet ćemo sa najjednostavnijim primjerom kako bi približili način djelovanja algoritama dinamičkog programiranja i njihovu praktičnu metalografsku primjenu. Naizgled jednostavni primjer mjerenja debljine prevlake zapravo i jest jednostavan ako se radi o tzv. OIM (Operator Interactive Measurements). Kod takvog mjerenja bilo bi dovoljno da operator (laborant) postavi početnu i krajnju mjernu točku na gornji i donji rub prevlake i da se ovisno o povećanju preračuna udaljenost mjernih točaka i odredi debljina mjerenog sloja. Međutim često se pojavljuje potreba za automatiziranim mjerenjima (AM), kao primjerice u laboratorijima proizvodnih pogona kod kojih treba provjeriti veliki broj kontrolnih uzoraka u jednom danu. Osim toga automatizirana mjerenja eliminiraju greške koje unosi ljudski faktor (laborant) jer nije uvijek lako i intuitivno točno označiti rubnu liniju prevlake ako se radi o nepravilnim konturama prevlake.



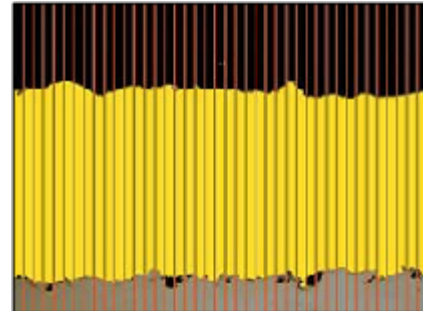
Slika 9 – početna slika uzorka [8]

Slika 9 prikazuje dio početne slike uzorka, a da bi se moglo početi sa kvantifikacijom problema najprije je potrebno provesti obradu slike kako bi odbacili nepotrebne informacije koje nemaju značaja za naše mjerenje a mogle

bi otežavati analizu. Da bi dobili uniformnu površinu koju je lakše analizirati moramo provesti tzv. thresholding. U ovom slučaju provodi se thresholding žutom ravninom koji je najlakše opisati kao provlačenje žute ravnine između dijela slike kojeg želimo analizirati (taj dio ostaje ispod ravnine) i ostatka slike. Desni dio koji prikazuje Slika 10 dijalog softvera za analizu na kojem se vidi kako je moguće regulirati gornji i donji prag osjetljivosti. Nakon thresholdinga koristi se operacija "Fill and Close" da bi rezultat thresholdinga pretvorili u uniformno žuto polje.



Slika 10 – thresholding žutom ravninom [8]



Slika 11 – dodavanje mreže [8]

Slika 11 prikazuje rezultat operacije "Fill and Close" nakon što je dodana i mreža koja dijeli prevlaku na uske pruge koje ćemo analizirati kao podprobleme dinamičkog programiranja.

Problem se pristupa na način da se izmjeri visina svakog pojedinog stupca i zatim se konačno rješenje izračuna na osnovu rezultata podproblema. Jedna od prednosti ovakvog pristupa je i to što rezultat mjerenja nije ograničen samo na debljinu prevlake, nego na osnovu analize rasipanja rezultata visine pojedinih stupaca možemo saznati i koliko je debljina prevlake ujednačena (što je često važno koliko i sam iznos debljine).

Ovaj primjer pokazuje pristup tipičan za računalnu analizu mikrostrukture i dobro ilustrira prednosti i fleksibilnost u odnosu na manualni postupak mjerenja. Osim toga, ovdje se radi o najjednostavnijem slučaju mjerenja koje se dovoljno jednostavno može provesti i manualno. Lako je zamisliti koliko tek ovakav pristup olakšava mjerenje nekih drugih svojstava koja zahtijevaju naporno prebrojavanje točaka i zatim uvrštavanje u odgovarajuće formule i računanje konačnog rješenja (npr. kod određivanja veličine zrna ili udjela faza).

Mjerenje veličine zrna

Kod konvencionalnog određivanja veličine zrna mjerenje se provodi prema standardu ASTM 112⁵ koji propisuje uvjete pod kojima se mjerenje obavlja, proceduru mjerenja i postupak prema kojem se izračunavaju rezultati. Kad se radi o računalnoj analizi mikrostrukture ista standardizacijska institucija (ASTM – American Society for Testing and Materials) postupak mjerenja definira standardom ASTM 1382⁶. Naravno, nije samo postupak računalnog mjerenja veličine zrna standardiziran. Vrlo veliki broj metalografskih mjerenja koja su standardizirana za manualne metode analize s vremenom je dobio svoju moderniju inačicu za primjenu u računalnoj metalografskoj analizi.

S obzirom na to da znamo na kojem principu rade algoritmi za automatsku detekciju objekata lako možemo zamisliti kako bi trebao izgledati postupak za određivanje veličine zrna. Nakon što softver (pomoću dinamičkog programiranja) odredi granice zrna moguće je analizirati uzorak na više načina. Jedan od logičnijih je taj da se jednostavno izračuna površina zrna i tada se odredi srednja vrijednost površina svih zrna. Ali tu dolazimo do problema sa graničnim zrnima jer će zrna na rubu slike imati manju površinu od realnog stanja. Zato je potrebno osmisliti neki način da se ova pojava anulira. Prema preporukama ASTM-a koristi se metoda slična metodi kruga kod manualnog određivanja veličine zrna. Umjesto da se analizira cijela slika u obzir se uzimaju samo ona zrna koja se nalaze unutar referentnog kruga (uključujući i ona zrna koja presijeca granična kružnica). Prema ASTM E:1382 moguće je koristiti i neke druge metode odabira zrna koja će se analizirati kao što su metoda više koncentričnih krugova, dijagonalnih linija, horizontalnih ili vertikalnih linija. Nakon što je na odgovarajući način odabran određeni broj zrna moguće je odrediti njihove površine (uz poznavanje povećanja slike), i tada iz tih podataka odrediti veličinu zrna.

Osim same veličine zrna koja proizlazi kao statistička veličina (najčešće aritmetička sredina, ali moguće je i vršiti korekcije u slučaju da nam se na referentnom dijelu slike pojavi atipično veliko ili malo zrno) softver za automatsku analizu slike će izračunati i neke druge zanimljive podatke. Moguće je statistički prikazati razdiobu veličine zrna, njihove odnose, tipični oblik i orijentaciju itd. Za neke metale važna je i mjera kružnosti zrna (npr. kod nekih materijala je za mehanička svojstva presudna veličina i što bolja kružnost zrna – nodularnost).

Određivanje udjela faza

Za sustave automatske analize slike (QIA) ovo je zapravo vrlo jednostavno mjerenje. Nakon pripreme slike i thresholdinga zadatak softvera je jednostavno

⁵ ASTM E112: determining Average Grain Size [9]

⁶ ASTM E1382: determining Average Grain Size using Semiautomatic and Automatic Image Analysis [9]

prebrojati broj piksela koji reprezentiraju traženu fazu i zatim izračunati postotak prebrojenih piksela u odnosu na sveukupni broj piksela na slici. Situacija zapravo nije puno kompliciranija ni kad je potrebno odrediti udjele faza kod materijala koji imaju više od dvije faze. Jednostavno se kod pripreme slike pazi na to da se svaka od faza dovoljno razlikuje od ostalih da bi je softver mogao točno detektirati, i dobit ćemo rezultate koji su za nekoliko redova veličina točniji od onih dobivenih (napornim i sporim) manualnim određivanjem udjela faza.

Zaključak

Prednosti računalne kvantitativne analize mikrostrukture (QIA) je zapravo vrlo teško kompletno sagledati, a pogotovo ih racionalno nabrojiti. Najveća prednost je svakako neizmjerljivo brži i lakši postupak mjerenja nekih svojstava mikrostrukture. Razliku između manualnog određivanja veličine zrna i nekoliko jednostavnih klikova mišem je teško pojmiti. A radi se samo o jednom mjerenju jednog svojstva. Razlika postaje drastična u slučaju da je vrijeme mjerenja ograničeno (ili je važna preciznost rezultata). Uz pomoć QIA moguće je mjerenja za koja je prije trebalo i po nekoliko sati provesti za nekoliko sekundi.

Ali ta ušteda na vremenu je tek vrh ledene sante. Sa prosječnim sustavom za automatsku analizu slike moguće je napraviti ogroman broj dodatnih mjerenja koja je manualnim metodama bilo nemoguće (ili vrlo teško) izvesti. Primjerice, manualnom metodom je bilo moguće odrediti veličinu zrna uzorka (i to je trajalo 15 – 30 min), dok uz pomoć sustava automatske analize rezultat veličine zrna dobijemo za manje od minute (uključujući vrijeme pripreme slike). Manualna metoda je davala samo jedan broj kao mjeru veličine zrna, dok QIA sustav nakon mjerenja veličine zrna nudi još i ogroman broj dodatnih informacija vezanih uz veličinu zrna. Vrhunski softverski QIA paketi za bilo koju vrstu nude u prosjeku po 250 dodatnih mjerenja i informacija. Skoro sve metode mjerenja i analize koje su objavljene i dokumentirane u literaturi su na ovaj ili onaj način implementirane u takve softverske pakete i dostupne kroz intuitivno i fleksibilno sučelje. Osim toga, QIA softveri jako olakšavaju izradu laboratorijskih izvješća.

Progresivni razvoj računala se odražava i na ovo softversko područje staro skoro 45 godina. U početku se veliki dio automatske analize bazirao na specijalnim elektroničkim sklopovima koji su imali zadaću prilagođavati i korigirati sliku, dok se razvojem procesne moći računala prešlo na fleksibilniju varijantu softverske korekcije. Daljnji zamah QIA sustavima je dao razvoj relativno jeftinih digitalnih kamera i fotoaparata koji su u mnogome olakšali digitalizaciju slike mikrostrukture (koja je dugo bila usko grlo procesa što se tiče trajanja postupka). Jedno vrijeme su se QIA sustavi mogli nabaviti jedino u paketu sa specijaliziranim kamerama i sustavima prijenosa i digitalizacije slike, ali trenutno svi sustavi mogu obrađivati digitaliziranu sliku bez obzira na njezin izvor. Za analitički softver čak nije ni važno radi li se o mikrostrukturi nekog uzorka, nekom medicinskom odnosno kemijskom uzorku ili čak satelitskom snimku. Softver je napravljen na način da prepoznaje i kvantificira određene strukture piksela i zatim prezentira njihove značajke. Naravno da je važno imati analitički softver prilagođen potrebama metalografskog laboratorija, ali većinu mjerenja je moguće obaviti na bilo kojem drugom QIA softveru uz malo više truda i vremena.

S vremenom i cijene QIA sustava postaju sve niže, tako da većini metalografskih laboratorija ovakav sustav postaje opravdana i lako isplativa investicija.

Popis korištene literature

- [1] <http://www.planck.de/english/illustrationsDocumentation/documentation/europWhiteBook/index.html>
- [2] http://en.wikipedia.org/wiki/Microscope_image_processing
- [3] http://en.wikipedia.org/wiki/Image_analysis
- [4] http://en.wikipedia.org/wiki/Dynamic_programming
- [5] <http://pubs.acs.org/cen/metallography/>
- [6] <http://www.zeiss.com/micro>
- [7] http://www.buehler.com/technical_information/Buehler%20Book/book.htm
- [8] <http://www.struers.com/>
- [9] <http://www.astm.com/>
- [10] <http://www.bioss.ac.uk/staff/chris/skagen.pdf>
- [11] <http://www.bioss.ac.uk/staff/chris/avtrw.pdf>
- [12] <http://www.reindeergraphics.com/>
- [13] Advanced Engineering Mathematics; E. Kreyszig; Wiley & Sons (1993)
- [14] "Adobe Photoshop CS2" sustav pomoći