

Odkrivanje napak v industrijskih tiskovinah s pomočjo slikovne poravnave

Jurij Rakun, Damjan Zazula

Fakulteta za elektrotehniko, računalništvo in informatiko
Smetanova 17
2000 Maribor
E-pošta: jurij@mozirje.com, zazula@uni-mb.si

Povzetek. V članku opisujemo postopek, ki smo ga razvili za potrebe industrijskega preverjanja kakovosti tiskovin. Postopek računalniško poravna sliki dveh tiskovin, in sicer referenčne ter primerjane. Poravnavo opravimo v dveh fazah: najprej grobo, nato še natančnejšo. Prva faza temelji na prilagodljivi binarizaciji obeh slik in določanju njunih težišč. Poravnavi težišč sledi še lokalno iskanje najboljšega prileganja, ki ga ovrednotimo z minimalno kvadratno razdaljo med vrednostmi vseh slikovnih elementov. Tako doseženo grobo ujemanje izboljšamo še z iteracijo afinih transformacij in vmesnih interpolacij obeh obravnavanih slik. Tak način detajlnega poravnavanja vpelje elastično korekcijo, tako da optimalno prekrije tudi slike, ki se nekoliko razlikujejo po velikosti ali strukturi. Poravnani sliki nato uporabimo za ugotavljanje odstopanj med istoležnimi slikovnimi elementi. Najočitnejša odstopanja opredelimo kot tiskarske napake, pri čemer se lahko omejimo le na sivinske slike, lahko pa upoštevamo tudi barvno informacijo. Opisani pristop je robusten in uspešno odkriva tiskarske napake tudi pri zahtevni industrijski embalaži. Pri stoodstotni pozitivni napovedljivosti doseže za dokumente s preprostejšimi motivi 95-odstotno senzitivnost, za zahtevne dokumente z veliko besedila pa okoli 90-odstotno.

Ključne besede: računalniška obdelava slik, poravnava slik, affine transformacije, pragovna segmentacija

Error Detection in Industrial Printed Matters Using Image Registration

Extended abstract. In this paper we describe a new approach to detect inferior quality and errors in printing matter using a regular PC and document scanner. Our method relies on a comparison of an inspected document with its referential version. It comprises coarse and fine document image registration first, and then detection of discrepancies between the aligned pixels or regions. Registration deploys binarized images and coarsely aligns them by their centres of gravity—Eq. (2). After overlaying the calculated centres, we match the images by rotating and translating them in a locally limited way. An optimum position is determined using the mean square error (MSE) of matched pixels—Eq. (3). The second registration step refines this position and is based on affine transformations which, utilizing the MSE criterion again, compute the rotation and translation for an optimum image match—Eqs. (4) to (8). This fine registration works iteratively, with interim interpolations, and introduces a kind of elastic image

correction and results in an optimum fit also when the proportions and sizes of the compared images are not entirely equal. After the images have been optimally registered, discrepancies between individual pixels or regions are looked for. Finally, distribution of detected differences is used in order to mark the most probable ones as caused by inferior printing quality. Errors can be searched either in gray or colour images, optionally also in separate colour planes. A prototype user interface is shown in Fig. 1.

We confirmed experimentally that the developed algorithms perform satisfactorily. The error detection rate in documents with a simple structure, mostly images, is about 95%, whereas with complex documents containing a lot of text this figure is about 90% (Figs. 2 nad 3). The results of automated printing errors detection are depicted in Fig. 4 for a simple document and in Fig. 5 for a more complex textual document.

The described approach has been successfully validated in an industrial firm. The only drawback found is its rather high computational complexity of $O(n^2)$.

Keywords: image processing, image registration, affine transformation, thresholding segmentation, image comparison

1. Uvod

Kljub vse večji uporabi elektronskih dokumentov je v številnih primerih tisk še vedno nujen. Spomnimo se samo tiskovin, kot so serijsko tiskana uporabniška navodila, slike, nalepke in podobni dokumenti, ki so pridruženi komercialnim izdelkom. Teh pri današnjem stanju prodaje in uporabe artiklov ni mogoče zamenjati z elektronsko zabeleženimi informacijami. Takoj pa, ko imamo opravka s številnimi ponovnimi tiskani serije dokumentov, ki se jim pri velikoserijski proizvodnji nikakor ne moremo izogniti, je zelo verjetno, da bo prihajalo do tiskarskih napak.

Odkrivanje tiskarskih napak je za človeka naporno in delikatno opravilo. Še posebej je zahtevno pri velikoserijski proizvodnji, saj na primer multinacionalka s področja kozmetične industrije samo v Sloveniji na leto uporabi več kot 350 milijonov tiskovin, med katerimi je treba pravočasno odkriti 1 do 2 % neuporabnih.

Na trgu obstaja veliko naprav, ki so namenjene za nadzor industrijskih procesov. Glede na način zajemanja slik jih lahko razdelimo v dve skupini. Prve zajemajo slike s kamero in jih najdemo predvsem v proizvodnih procesih [7, 9]. Tu skrbijo za pravilen potek procesa, kot je npr. polnjenje steklenic, pakiranje proizvodov ipd. Druge, namenjene predvsem za preverjanje tiskovin, pa temeljijo na uporabi namenskih prebirnih naprav [8]. Vsem je skupno, da pri različnih zajemih slik v digitalizirano obliko delujejo v nadzorovanih pogojih, saj slike zajemajo pri isti osvetljenosti in predvsem na istih pozicijah [9]. S tem se postopek odkrivanja odstopanj močno poenostavi.

Bistvena razlika med navedenimi pristopi in zahtevami, ki se jim podreja rešitev, opisana v tem članku, je, da ob slikah nimamo natisnjenih oznak, po katerih bi lahko slike medsebojno pravilno prekrili. Takšen način poravnavanja slik je trivialen, naša rešitev pa mora za poravnavo poskrbeti z bolj domišljenim pristopom, ki deluje na samih slikah, seveda neodvisno od njihove vsebine. Na ta način lahko realiziramo računalniško iskanje tiskarskih napak, ne da bi potrebovali predrago, specializirano videoopremo, izognemo pa se tudi neposrednemu preverjanju, pri katerem so etikete že nalepljene na izdelku. Taka rešitev je celo nezaželeno, saj je treba izdelek, ki ima nalepljeno pokvarjeno etiketo, v celoti izločiti, čeprav z njim samim ni nič narobe.

V nadaljevanju bomo opisali postopek, ki išče tiskarske napake tako, da pozna referenčno tiskovino in z njeno sliko primerja sliko preverjanega dokumenta. V drugem poglavju opisujemo izhodiščne zahteve ter ideje

za ustrezno računalniško poravnavanje slik in iskanje odstopanj med njimi. Tretje poglavje je namenjeno uporabniškemu vmesniku in kratki razlagi posameznih funkcij, ki jih ponuja uporabniku. Rezultati delovanja izdelanega prototipnega programa za iskanje tiskarskih napak so prikazani in ovrednoteni v četrtem poglavju, peto poglavje pa razpravlja o prednostih in slabostih uvedene rešitve ter sklene članek.

2. Avtomatizirano računalniško iskanje tiskarskih napak

S prototipnim računalniškim programom poskušamo predvsem poenostaviti in avtomatizirati človeško delo, ki je sicer edina možnost za preverjanje pravilnosti tiskovin. V nasprotju z napravami, namenjenimi analizi tiskovin in pravilnosti postopkov v proizvodnji, ki večinoma opravljajo 100-odstotno preverjanje, je naš prototip namenjen predvsem vzorčnemu preverjanju. Tak pristop pospeši, nekoliko poenostavi in posploši proces odkrivanja napak in je po izkušnjah velikoserijskih proizvajalcev dovolj zanesljiv.

Želimo, da naš program omogoča univerzalno analizo odstopanj med dvema slikama. Omogoča naj analizo pri različnih ločljivostih slik, pri različni poravnosti motivov slik in celo pri slikah, ki niso čisto enako velike. Od znanih komercialnih naprav se razlikuje tudi po tem, da omogoča analizo napak na različni strojni opremi (ali s prebirniki ali z digitalnimi kamerami) in za različne tiskovine (napise na embalažnih škatlah, prozorne in neprozorne nalepke, navodila ipd.).

Kakor smo povedali že uvodoma, je postopek poravnave slik eden glavnih korakov naše rešitve. Da dosežemo najboljšo poravnost slik, uporabljamo postopke za togo in elastično poravnavo slik. Prvi temeljijo na iskanju ter poravnavi težišč binariziranih slik [1] in so uporabljeni za grobo poravnavo. Drugi, torej postopek elastične poravnave [2], pa poskrbi še za fino poravnavo in je znan na področju obdelave medicinskih slik, npr. za magnetnoresonančne naprave.

Ko sta referenčna in primerjana slika dovolj dobro poravnani, lahko primerjamo istoležne slikovne elemente in ugotavljamo, kje nastajajo območja morebitnih napak, torej odstopanj, ki niso posledica obdelovalnih postopkov in zaokrožitvenih napak. Treba se je odločiti, kako veliko naj bo območje z napako, da bomo nanj opozorili. Ni se treba, denimo, obremenjevati z napakami, ki jih s prostim očesom niti ne vidimo. Za preverjanje serijskih nalepk v industriji je dovolj poiskali le napake, ki so večje ali enake 0,2 mm [11]. Premer iskanega napačnega območja torej lahko izrazimo v slikovnih elementih takole:

$$d \geq \frac{L}{25,4} 0,2 \text{ pikslov}, \quad (1)$$

kjer pomeni L ploskovno ločljivost slike, podano v številu slikovnih elementov na palec (25,4 mm). Pri ločljivosti 300 pik na palec, denimo, znaša izračunani premer 3 slikovne elemente. Pri odkrivanju napak v slikah z navedeno ločljivostjo se bomo torej osredotočili na iskanje odstopajočih območij, katerih premer je vsaj 3 slikovne elemente.

Poravnava referenčne in primerjane slike mora odstraniti tako morebitni zasuk kot premik med njima. Zato smo jo zasnovali v dveh korakih: najprej s hevrstičnim pristopom primerjani sliki poravnamo grobo, v drugem koraku pa ju poskušamo optimalno prekriti z afino transformacijo.

2.1 Groba poravnava

Grobo poravnavo slik smo izvedli s poravnavo težišč [1], ki ju izračunamo v binarizirani referenčni in primerjani sliki.

Binarizacijo slik opravimo z uporabo histogramov, in sicer tako, da izločimo ozadje in poskušamo ohraniti vse druge objekte na sliki. Velikokrat se izkaže, da vrednosti v histogramu niso unimodalne in da izločanje ozadja ni trivialno. Lotili smo se ga z iskanjem histogramskega maksimuma, ker predvidevamo, da je slikovnih elementov, ki imajo barvo ozadja, največ. Od maksimuma se spuščamo v obe smeri in iščemo minimalni vrednosti – torej dolini levo in desno od maksimuma. Najdeni minimumi so prag za izločanje ozadja. Ker pa ti minimumi ponavadi niso izraziti, postavimo prage šele po izvedbi iterativnega iskanja najboljše pozicije. Postopek deluje tako, da prvi potencialni prag postavimo pri prvem najdenem minimumu. Potem gledamo, ali se po premiku za en slikovni element histogram morda spet spusti. Če se, iščemo še drugi, bolj oddaljeni minimum in tam postavimo drugi prag. Postopek nadaljujemo do največ petega potencialnega praga.

Z vsakim od postavljenih potencialnih pragov za izločanje ozadja sliko binariziramo in ji določimo težišče. Enako seveda naredimo tako za referenčno kot za primerjano sliko.

Ko sta sliki binarizirani, izračunamo težišči po formuli (2), kar naredimo po vsaki iteraciji binarizacije posebej:

$$x_{t_i} = \frac{\sum_{x=1}^M \sum_{y=1}^N b(x, y) \cdot x}{t_i}, \quad y_{t_i} = \frac{\sum_{x=1}^M \sum_{y=1}^N b(x, y) \cdot y}{t_i}, \quad (2)$$

kjer so: $t_i = \sum_{x=1}^M \sum_{y=1}^N b(x, y)$ število uporabljenih

slikovnih elementov za izračun težišča, $b(x, y)$ binarizirana slika, x in y koordinati, M število stolpcev in N število vrstic v sliki.

Sedaj težišči poravnamo in ocenimo trenutno ujemanje slik po formuli (3):

$$O = \sqrt{\sum_{x=1}^M \sum_{y=1}^N (p_1(x, y) - p_2(x, y))^2}, \quad (3)$$

kjer pomenijo O oceno ujemanja dveh slik, $p_1(x, y)$ sivinsko vrednost v sliki 1 in $p_2(x, y)$ sivinsko vrednost v sliki 2, obe na poziciji x, y .

Najboljše ujemanje (minimalni O) določa binarizirano verzijo primerjane slike, ki jo obdržimo za nadaljnjo obdelavo. Izračunani težišči zaokrožimo na celoštevilski vrednosti in sliki premaknemo tako, da se s težišči pokrivata. Izberemo si območje $\pm 5^\circ$, v katerem bomo s sukanjem primerjane slike proti referenčni poiskali boljše ujemanje. Da zmanjšamo zahtevnost postopka, najprej poskušamo prilagajanja s korakom 5° v območju $[-5^\circ, 5^\circ]$. Pozicijo z najmanjšim odstopanjem slik obdržimo, okoli nje pa poiščemo še boljše ujemanje ob zasukih za 1° v intervalu $[-4^\circ, 4^\circ]$. V rotacijo primerjane slike je seveda vključena interpolacija, ki dodatno povečuje računsko zahtevnost postopka.

Za tem poskušamo sliki na hitro še bolje poravnati s translacijami v bližnji okolici najboljše pozicije, določene s sukanjem. Premikamo se najprej za dva, nato pa še za en slikovni element v vse smeri, dokler uspevamo najti kakšno boljše ujemanje (manjši O).

Groba poravnava sliki navadno že zelo dobro prekrije, vendar še ne toliko, da bi lahko odstopanja med njima opredelili iz razlik med vrednostmi istoležnih slikovnih elementov. Zato moramo poseči še po optimizaciji z afino transformacijo. Temu bomo preprosto rekli fina poravnava.

2.2 Fina poravnava

Uporabimo postopek poravnave slik iz [2], ki temelji na iteracijah z afinimi transformacijami in vmesnimi interpolacijami slik. S tem slike dodatno poravnamo, tako da se bodo, kolikor je le mogoče, ujele. Uporaba rekurzivnih afinih transformacij pa vpeljuje še dodatno elastično korekcijo slik, ki sliki geometrijsko popravi, če med njima obstajajo manjša geometrijska neskladja.

Po [2] izpeljemo postopek fine poravnave na naslednji način. Vzemimo, da sta ekvivalentna slikovna elementa iz prve in druge slike označena kot $p(x, y, t)$ in $p(x, y, t-1)$. Parameter t določa referenčno, parameter $t-1$ pa primerjano sliko. Če gre za sliki z enakim motivom, lahko slikovne elemente ene slike izrazimo tako, da ustrezno transformiramo lego slikovnih elementov iz druge slike. To zapišemo kot:

$$p(x, y, t) = p(m_1x + m_2y + m_5, m_3x + m_4y + m_6, t-1). \quad (4)$$

Parametri od m_1 do m_6 v enačbi (4) so v tem primeru koeficienti, ki jih bomo določili z uporabo affine transformacije. Zapišimo sedaj oceno trenutne poravnave kot vsoto kvadratov razlik med istoležnimi slikovnimi elementi za primer, ko sta sliki veliki $M \times N$ slikovnih elementov:

$$E(\vec{m}) = \sum_{x=1}^M \sum_{y=1}^N (p(x, y, t) - p(m_1x + m_2y + m_5, m_3x + m_4y + m_6, t - 1))^2, \quad (5)$$

pri čemer pomeni $\vec{m} = (m_1, m_2, m_3, m_4, m_5, m_6)^T$ vektor afinih parametrov.

Afini parametri m_i , ki jih iščemo, v enačbi (5) ne nastopajo eksplicitno. Zato enačbo razvijemo v Taylorjevo vrsto in jo aproksimiramo s členi prvega reda:

$$E(\vec{m}) \approx \sum_{x=1}^M \sum_{y=1}^N \{p(x, y, t) - [p(x, y, t) + [(m_1 - 1)x + m_2y + m_5]p_x(x, y, t) + [m_3x + (m_4 - 1)y + m_6]p_y(x, y, t) - p_t(x, y, t)]\}^2, \quad (6)$$

kjer p_x , p_y in p_t pomenijo parcialne odvode po x , y in t . Pri izvedbi postopka te odvode ocenimo z diferencami. Enačbo (6) poenostavimo in uvedemo vektorski zapis, kjer je $\vec{c} = (x \cdot p_x, y \cdot p_x, x \cdot p_y, y \cdot p_y, p_x, p_y)^T$ in skalar $k = p_t + x \cdot p_x + y \cdot p_y$. Zaradi preglednosti smo izpustili argumente pri parcialnih odvodih. Nova oblika zapisa je naslednja:

$$E(\vec{m}) = \sum_{x=1}^M \sum_{y=1}^N (k - \vec{c}^T \vec{m})^2. \quad (7)$$

S parcialnim odvajanjem enačbe (7) po \vec{m} pridemo do minimalne napake $E(\vec{m})$, če velja:

$$\vec{m} = \left[\sum_{x=1}^M \sum_{y=1}^N \vec{c} \vec{c}^T \right]^{-1} \sum_{x=1}^M \sum_{y=1}^N \vec{c} k. \quad (8)$$

Z dobljenimi parametri m_i izvedemo afino transformacijo, vendar pa se moramo zavedati, da omogoča le lokalno omejena, manjša približevanja k optimalni rešitvi. Zato jo izvajamo iterativno. Zaradi manjše časovne zahtevnosti uporabimo svinske slike. V vsaki iteraciji transformiramo vmesne slike iz prejšnjega koraka z novimi afinimi parametri \vec{m} .

Koordinate slikovnih elementov po transformaciji niso več celoštevilske, zato moramo slike interpolirati. Uporabljamo linearno interopolacijo v Moorovem sosedstvu z radijem 1 [6]. Izkazalo se je, da interpoliranje samo ene slike vnaša dodatne odstopanja med primerjanima slikama. Zato vsako interpolacijo opravimo tako na originalni kot na primerjani sliki: izračunane parametre afine transformacije razdelimo na zasuk in premik, polovico obeh apliciramo na eni sliki, polovico pa v nasprotni smeri na drugi sliki. Tako je skupni učinek enak, kot bi bil s celotno transformacijo samo na eni sliki, vendar pa interpolacijske spremembe (povprečenja v okolici slikovnih elementov) podobno vplivajo na obe primerjani sliki hkrati.

Ko odkrijemo najboljšo skupno poravnavo, iteracijski postopek zaključimo in uporabimo dobljene kumulativne parametre \vec{m} pri izvorni, neporavnani barvni sliki.

3. Računalniška realizacija z uporabniškim vmesnikom

V tej sekciji na kratko povzemamo, kako prototipni program, ki smo ga izdelali, pomaga uporabniku pri učinkovitem iskanju odstopanj med primerjanima tiskovinama. Slika 1 kaže videz uporabniškega vmesnika.

Začnimo z gumboma za nalaganje referenčne in primerjane slike tiskovin s pomnilnega medija (označena kot »slika 1« in »slika 2« zgoraj desno). Ker ne vemo, s kakšnim prebirnim sistemom bo program deloval, podpiramo vnos različnih formatov slik, kot so bmp, tiff, pcx, jpeg, gif itd. Priporočljiva je uporaba nekomprimiranih slik (npr. bmp), saj v nasprotju s komprimiranimi ne vsebujejo šuma, ki je posledica kompresije slik.

Nadaljujemo z grobo poravnavo. Kot smo spoznali v prejšnjem podpoglavju, naš program z grobo poravnavo kompenzira zasuk in premik, ki nastaneta, če sliki ob prebiranju nista popolnoma enako položeni na prebirno površino. Začetni zasuk lahko nastavimo na 0° , 90° , 180° ali 270° (privzeto 0°), tako da kliknemo v ustrezna okenca. V dodatnem pogovornem oknu lahko odredimo, v kakšnih kotnih premikih bomo poravnavali iz izbrane začetne lege (med $\pm 5^\circ$ in $\pm 45^\circ$).

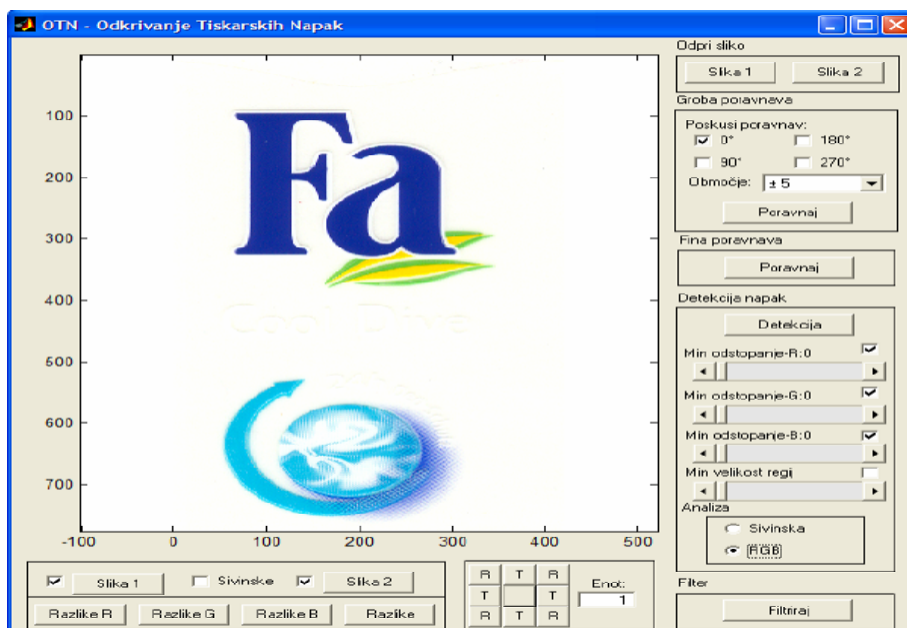
Sledi fina poravnava. Ta, kot vidimo na sliki 1, ne omogoča nobenih dodatnih nastavitvev in jo sprožimo po končani grobi ali ročni poravnavi slik (gumb »Poravnaj«).

Pod gumbi za poravnave so v uporabniški vmesnik vgrajeni drsniki za iskanje napak. Izberemo lahko sivinsko ali barvno analizo, tako da kliknemo krožca »Sivinska« ali »RGB« v polju »Analiza« (spodaj desno). Obe izbiri predvidevata dodatno nastavitvev praga za odkrivanje napak. Nastavimo ga s pomočjo treh drsnikov, ki sledijo gumbu »Detekcija«, za vsako posamezno barvno ravnino posebej.

S četrtem drsnikom določimo minimalne velikosti napačnih regij, v katere lahko združujemo odstopajoče sosednje slikovne elemente. Kadar želimo prikazovati napačne regije, moramo klikniti v polje nad drsnikom za velikost regij.

Gumb »Filtriraj« čisto desno na dnu uporabniškega vmesnika sproži obdelavo naloženih slik z nizkim sitom (slikovne elemente nadomesti s povprečji okolice 3×3). Sliki tako zgladi in s tem odstrani drobne napake, ki so posledica šuma. Filter je programu dodan kot opsijski korak, ki ga lahko izvedemo pred iskanjem napak.

Na dnu uporabniškega vmesnika najdemo devet gumbov za ročno poravnavo. Gumbom je pridruženo vnosno polje »Enot«. V slednje vpišemo velikost



Slika 1: Prikaz uporabniškega vmesnika prototipne rešitve programa za iskanje tiskarskih napak
Figure 1: User interface for a prototype of detection of printing errors

koraka, s katerim želimo izvajati ročno poravnavo. Pri zasuku so to stopinje, pri premiku pa slikovni elementi. Ročno poravnavo moramo najprej aktivirati, kar postorimo s sredinskim gumbom. Z njim ročno poravnavo tudi sklenemo (deluje torej po principu vklop/izklop). Zgoraj, spodaj, levo in desno so gumbi, ki so namenjeni premikom gor, dol, levo in desno. Označeni so z napisom »T«. Diagonalno so nameščeni gumbi z oznakami »R«. Le-ti omogočajo ročne rotacije. Gumb, ki je levo od gumba zgoraj, rotira v negativno smer, gumb pa, ki je desno od gumba zgoraj, rotira v pozitivno smer. Dodatna gumba sta namenjena hitrejšemu sukanju, kadar sta primerjani sliki zelo zasukani. Levi opravlja zasuk za 90°, desni pa za 180°. Oba sučeta v pozitivno smer.

V levem spodnjem vogalu uporabniškega vmesnika srečamo še večje področje, ki je razdeljeno na dve podpodročji. Zgornje je namenjeno prikazu slik, spodnje pa prikazu odstopanj. Polji ob gumbih »Slika 1« in »Slika 2« določata, katera slika bo prikazana na delovni površini uporabniškega vmesnika (obe, ena izmed obeh ali nobena). Vmes najdemo še dodatno polje, ki omogoča prikaz slik v barvah ali s sivinami. Vsako sliko pa lahko s klikom na ustrezno tipko »Slika« pogledamo še v povečavah. Odpre se namreč dodatno okno, v katerem je omogočeno povečevanje in zmanjševanje slik (zoom).

Spodnje podpodročje je sestavljeno iz štirih gumbov. Prvi trije so namenjeni prikazu razlik (napak) med barvnima slikama. Zadnji, skrajno desni pa ima dve funkciji. Prva je prikaz združenih napak v vseh treh barvnih ravninah, če smo opravili barvno analizo, druga

pa je prikaz napak v primeru sivinske analize, ko imamo samo eno, sivinsko ravnino.

4. Rezultati računalniškega odkrivanja tiskarskih napak

Uporabnike programa za avtomatično odkrivanje tiskarskih napak zanimajo predvsem najočitnejša odstopanja na preiskovanih tiskovinah, zato nima smisla, da bi jim takoj pokazali vse odkrite napake. Predvidevamo, da so vrednosti odstopanj med primerjanima slikama normalno porazdeljene, zato izračunamo prag, ki odredi, katera odstopanja bodo prikazana, takole:

$$D = \mu + 3\sigma, \quad (9)$$

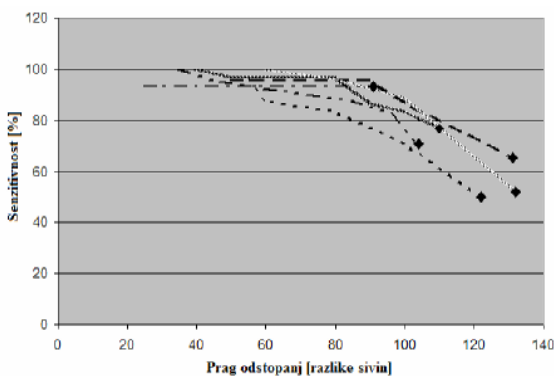
pri čemer so D vrednost praga, μ povprečna vrednost odstopanj med slikama in σ njihov standardni odklon. Pri normalni porazdelitvi je 99,74 odstotkov vseh odstopanj pod pragom iz enačbe (9), kar se je v praksi izkazalo kot ugodno privzeto vrednost. Seveda pa lahko uporabnik to nastavitev pozneje ročno tudi spremeni (poveča ali zmanjša).

Za preizkušanje vseh faz delovanja razvitega prototipnega programa za odkrivanje tiskarskih napak smo uporabili dva pristopa. V prvem smo vzeli slike tiskovin, ki smo jim umetno dodali naključna odstopanja. Tako smo pri odkrivanju napak točno vedeli, katere smo zares uspeli odkriti in katerih ne. Drugo skupino poskusov pa smo izvedli s pari realnih slik, pri katerih smo sami na pogled ocenili, kje se razlikujejo. Potem smo rezultate računalniškega odkrivanja napak primerjali s svojimi odčitki.

4.1 Validacija postopka z referenčnimi slikami in umetno dodanimi napakami

Naključno smo izbrali nekaj slik iz nabora, ki so nam ga posredovali iz Henkla Slovenija d. o. o. v Mariboru. Vsako od njih smo v paru (identični sliki) posredovali našemu analiznemu programu, pri tem pa smo eno od slik v paru umetno pokvarili z dodajanjem različnih tipov odstopanj. Sliko smo pokvarili tako, da smo ji na naključnih mestih dodali naključno število napak, vendar ne več kot 30. Te napake so naključne velikosti (od 1×1 do 10×10 slikovnih elementov) z naključno izbranimi dvema tipoma odstopanj. Za prvo vrsto umetno dodanih odstopanj je značilna popolna neodvisnost od predela slike, kamor so umeščena. O drugem tipu napak pa govorimo, ko se barvi spremenjenega slikovnega predela in umetno dodanih odstopanj sicer ujemata, a pride do razlik v odtenkih. Kakšen tip napake bo izbran za vsako od dodanih umetnih napak, je določeno naključno, tako da je 50 % dodanih napak prvega, 50 % pa drugega tipa. Pri obeh tipih je za vsako barvno ravnino mogoče odstopanje v intervalu [0, 255], le da pri prvem tipu barvo izberemo naključno, pri drugem pa jo določajo prvotne barve spreminjanega slikovnega območja.

V vseh poskusih smo uspešnost avtomatičnega

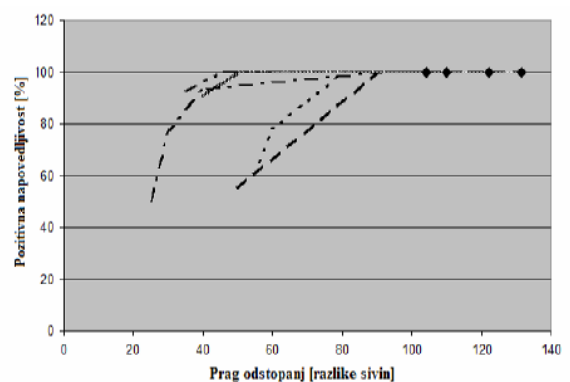


odkrivanja tiskarskih napak ugotavljali s pomočjo senzitivnosti in pozitivne napovedljivosti. Senzitivnost podaja odstotek pravilno razpoznanih napak, pozitivna napovedljivost pa določa odstotek resničnih napak izmed vseh javljenih napak. Rezultate grafično prikazujeta sliki 2 in 3. Na obeh sta leva grafa namenjena prikazu senzitivnosti, desna pa pozitivni napovedljivosti. Rezultati s slike 2 so bili dobljeni z analizo slik, ki imajo relativno preproste motive in le malo besedila, medtem ko so rezultati s slike 4 dobljeni pri analizi kompleksnejših slik, na katerih je veliko besedila (gre za proizvajalčeva uporabniška navodila).

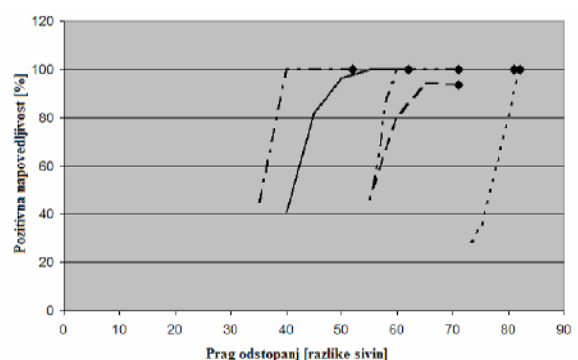
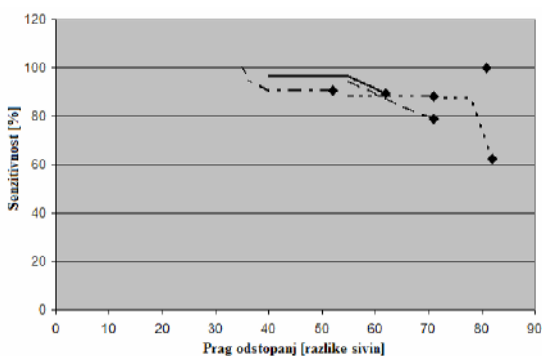
Z znakom karo smo v grafih na slikah 2 in 3 prikazali točko, ki ustreza avtomatični začetni nastavitvi praga za odkrivanje napak in pomeni približno 0,3 % najočitnejših napak, torej največjih slikovnih odstopanj.

4.2 Rezultati analize tiskovin s tiskarskimi napakami

Iz nabora industrijskih tiskovin firme Henkel iz Maribora smo izbrali pare slik, med katerimi je bila ena brez napak, torej referenčna, druga pa je vsebovala odstopanja, ki smo jih lahko zaznali že s prostim očesom.



Slika 2: Senzitivnost (levo) in pozitivna napovedljivost (desno) za slike s preprostimi motivi
Figure 2: Sensitivity (left) and positive predictivity (right) for images with simpler motives



Slika 3: Senzitivnost (levo) in pozitivna napovedljivost (desno) za slike z zapletenimi (tekstovnimi) vsebinami
Figure 3: Sensitivity (left) and positive predictivity (right) for images with more complex (textual) contents

Slika 4 kaže primer analize za nalepko »Fa«. Levo so z rdečimi krožci označena mesta, za katera smo opazili odstopanja v tisku. Desno je prikazan rezultat avtomatičnega iskanja napak na sivinski sliki barvnega originala »Fa«. Tudi naš program najdene napake obkroži rdeče. Vidimo, da so bila najdena vsa odstopanja, ki smo jih prej opazili že sami, poleg tega pa je avtomatična analiza odkrila še precej dodatnih slikovnih elementov, pri katerih je razlika med referenčno in primerjano sliko večja od nastavljenega praga. V danem primeru smo uporabili privzeto vrednost praga, ki izloči približno 0,3 % najočitnejših napak.

Podobne rezultate prikazuje slika 5 za embalažo proizvoda »Silk-Flex«. Tudi tukaj so v levem delu označene napake, ki smo jih opazili s prostim očesom. Na sredini so na sivinski sliki prikazani rezultati avtomatične analize. Vidimo, da je program že pri

visokem privzetem pragu za 0,3 % največjih napak odkril množico odstopanj, ki jih prej sami nismo zaznali. Za industrijskega uporabnika je tako obsežno opozarjanje na množico drobnih napak, ki se lahko povezone z enim samim slikovnim elementom, celo moteče. Naš prototipni program ponuja dve rešitvi: ali dvigne prag za zaznavo napak ali pa zahteva razvrščanje napak v regije. Prvi način sicer reducira število prikazanih napak, pri čemer pa lahko izločimo tudi takšne, ki so očitne in bi ne smele biti spregledane. Drugi način, pri katerem se odločimo, kako veliko mora biti združeno napačno območje, da ga prikazemo, je potencialno boljši, saj nikoli ne izpusti očitnih napak. Združevanje napak v regije privede tudi do nekoliko drugačnega prikaza: program združena napačna območja uokviri, medtem ko posamezne napake obkrožuje, kakor smo že omenili. Na sliki 5 desno je prikazano, katere napake označi avtomatična analiza, če



Slika 4: Primer analize za motiv »Fa«: napake, najdene s prostim očesom (levo), in napake, najdene s pomočjo programa (desno)
 Figure 4: Printing error detection for label "Fa": errors as seen by a human observer (left), and found by our algorithm (right)



Slika 5: Primer analize za motiv »Silk-flex«: napake, najdene s prostim očesom (levo), in posamezne napake, najdene s pomočjo programa (sredina) ter združene v regije (desno)
 Figure 5: Printing error detection for document "Silk-flex": errors as seen by a human observer (left), found by our algorithm (middle), and grouped into erroneous regions (right)

dovolimo le regije, ki presegajo velikost 3×3 slikovne elemente.

5. Razprava in sklepi

Kakor smo statistično pokazali v podpoglavju 4, so opisani računalniški postopki za avtomatično odkrivanje tiskarskih napak zanesljivi in natančni. Eksperimentalno smo ugotovili, da odkrijemo na slikah s preprostejšimi motivi okoli 95 %, na zahtevnih slikah z veliko besedila pa okoli 90 % resničnih napak, ne da bi pri tem javljali fantomske napake.

Prototip analiznega programa je že prestal preizkušanje v industrijskem okolju: pri Henklu Slovenija, d. o. o., v Mariboru so z njim več mesecev obdelovali svoje industrijske tiskovine. Očitali so mu dve pomanjkljivosti: preveliko časovno zahtevnost in preveliko občutljivost. Slednja seveda ni problematična, kakor smo že razložili v podpoglavju 4. Če uporabnik meni, da program odkriva napake, ki za opazovalca s prostim očesom niso moteče, lahko vedno zmanjša njegovo občutljivost, tako da poveča prag za prikaz napak. Resnejši pa je očitek o časovni zahtevnosti, ki je $O(n^2)$, če n pomeni dimenzijo slike. Izmerili smo, da lahko traja iskanje napak pri večjih tiskovinah, ki vsebujejo veliko besedila z ločljivostjo 300 pik na palec, tudi 15 minut in več. To je za serijske preizkuse v industriji nesprejemljivo. Vzrok seveda tiči v veliki računski zahtevnosti poravnavanja primerjanih slik. Kaj se da storiti?

Pri grobi poravnavi bi se vsekakor lahko zatekli v frekvenčni prostor. Vemo, da se iz faznega spektra da ugotovili premik, iz amplitudnega spektra pa še zasuk med slikama. S tem bi nekoliko skrajšali obdelavo, ne pa dovolj. Bolj problematična je namreč fina poravnava. Pri tej lahko poskušamo z zmanjšanjem števila uporabljenih slikovnih elementov. Eden od načinov je večločljivostni pristop, kjer za prvo iteracijo uporabimo sliko nekajkrat manjše ločljivosti in dobljene parametre poravnave prenesemo na sliko večje ločljivosti. Nato ločljivost izboljšamo in postopek ponovimo. Ponavljamo, dokler ne pridemo do slik izvornih ločljivosti.

Alternativni način pa predvideva, da namesto celotne slike v poravnavo vključujemo samo njene dele. Ker gre za toge slike, neujemanje enega dela slike pomeni neujemanje tudi pri drugih delih slike. Sliko lahko razdelimo, recimo, na četrte in v poravnavo vsako četrtno posebej drugo za drugo, parametre poravnave pa ob vsaki iteraciji upoštevamo za celotno sliko. Na koncu lahko še nekaj iteracij poravnave izvedemo za celotno sliko. Ta način se v praksi izkaže za najperspektivnejšega.

Predstavljena prototipna rešitev je izpolnila pričakovanja industrijskega partnerja, kar se tiče natančnosti in zanesljivosti. Prvi koraki v smeri nadaljnjih raziskav pa kažejo, da bo tudi z računalniki

današnjih zmogljivosti mogoče bistveno znižati trenutne časovne zahtevnosti postopkov za avtomatično iskanje tiskarskih napak.

Zahvala

Avtorja se zahvaljujeta univ. dipl. inž. Borutu Suhadolniku iz Henkla Slovenija d. o. o., ki je s svojimi nasveti in s preizkušanjem prototipne rešitve pomembno pripomogel k zasnovi predstavljenega postopka.

Literatura

- [1] D. Zazula, *Obdelava signalov in slik* (zapiski predavanj), 2005, neobjavljeno gradivo.
- [2] S. Periaswamy, H. Farid, "Elastic Registration in the Presence of Intensity Variations", *IEEE transactions on medical imaging*, let. 22, št. 7, str. 865–874.
- [3] R. P. Woods, "Spatial Transformation Models," v: *Handbook of Medical Imaging*, Isaac Bankman (ur.), Academic Press, San Diego, 2000, str. 465–497.
- [4] R.C. Gonzalez, *Digital image processing: using Matlab*, 1. izd., Pearson Prentice Hall, Upper Saddle River, 2003.
- [5] A. Bovik, *Handbook of Image and Video Procession*, 1. izd., Academic press, 2000.
- [6] M. Seul, *Practical Algorithms for Image Analysis: description, examples, and code*, 1. izd., Cambridge University Press, Cambridge, 2001.
- [7] QEA, *IAS-1000*, <http://www.qea.com/ias1000.html>
- [8] QEA, *Scanner IAS*, http://www.qea.com/pdf/datasheet_scannerias.pdf
- [9] FDS, *FDS Imaging Software*, <http://www.fdsresearch.com>
- [10] B. Suhadolnik: ustna navodila, Henkel Slovenija d. o. o., Maribor, junij 2005.

Jurij Rakun je leta 2005 diplomiral na Fakulteti za elektrotehniko, računalništvo in informatiko v Mariboru. Od takrat je na tej fakulteti vpisan na podiplomski študij računalništva in informatike. Maja 2006 je s temo, ki o obravnava pričujoči prispevek, zasedel drugo mesto na tekmovanju študentskih raziskovalnih člankov Regije 8 pri IEEE (Evropa, Afrika in Azija brez Daljnega vzhoda).

Dr. Damjan Zazula je diplomiral, magistriral in doktoriral na Fakulteti za elektrotehniko v Ljubljani. Za seboj ima 17 let razvojnega dela v industriji, od leta 1999 pa je redni profesor za računalništvo na Fakulteti za elektrotehniko, računalništvo in informatiko v Mariboru. Raziskovalno se posveča analizi in dekompoziciji biomedicinskih signalov in slik ter konceptualnim simulacijskim modelom za poučevanje in urjenje.