Универзитет "Св. Кирил и Методиј" – Скопје
Електротехнички Факултет

Систем за оптичко препознавање на македонски кириличен текст за помош на лица со оштетен вид

– магистерска работа –

ментор:
Проф. д-р Драган Михајлов

кандидат:
Дејан Ѓорѓевиќ

Скопје, 1997

University "Sv. Kiril i Metodij" - Skopje
Faculty fo Electrical Engineering

Dejan Ǵorǵevik

System for Optical Character Recognition of Macedonian Cyrillic Texts Aimed for Helping Visually Impaired Persons

- thesis -

Skopje, 1997

Ментор: проф. д-р Драган Михајлов
Електротехнички факултет -
Скопје

Комисија за одбрана:
Претседател:

Членови:

Датум на одбрана:

Датум на промоција:

Магистерскиот труд е од областа на
КОМПЈУТЕРСКАТА ТЕХНИКА И ИНФОРМАТИКА

Дејан Д. Ѓорѓевиќ, дипл. ел. инж.

Систем за оптичко препознавање на македонски кириличен текст за помош на лица со оштетен вид

АПСТРАКТ: *Во овој труд е прикажан систем за оптичко препознавање на печатен македонски кириличен текст наменет за помош на лица со оштетен вид. Тој е дел од поголем систем за помош на лица со оштетен вид кој вклучува и претворање на текст во говор и печатење со браево писмо. Препознавањето е базирано на сегментација и препознавање на изолирани знаци. Класификацијата е изведена со тежинско гласање на адаптивни логички мрежи. Изведено е и контекстно постпроцесирање на ниво на изолирани зборови со употреба на лексикон за македонскиот јазик.*

КЛУЧНИ ЗБОРОВИ: оптичко препознавање на знаци, адаптивни логички мрежи, комбинирање на класификатори, тежинско гласање, контестна обработка, компјутерски лексикон

Dejan D. Ǵorǵeviḱ, BS

System for Optical Character Recognition of Macedonian Cyrillic Texts Aimed for Helping Visually Impaired Persons

ABSTRACT: *System for optical character recognition (OCR) of Macedonian Cyrillic texts is presented in this thesis. It's aimed for helping visually impaired persons as a part of a larger system, which includes text-to-speech conversion for Macedonian language, and Braille printing. The recognition is based on character segmentation and isolated character recognition. Character classification is performed using weighted voting of adaptive logic networks. Contextual postprocessing on word level using a lexicon for Macedonian language is also performed.*

KEYWORDS: optical character recognition, OCR, adaptive logical network, ALN, classifier combining, weighted voting, contextual postprocessing, computer lexicon

Содржина:

1 Вовед 1

2 OCR системи 3

2.1 Историјат и поделба на писмата 3

2.2 Потреба од OCR 4

2.3 Историјат и поделба на OCR 6

2.4 Компјутерски систем за помош на слепи лица 8

3 Препознавање на документи 13

3.1 Автоматска интерпретација на документи 13

3.1.1 Препознавање на графика 13

3.1.2 Препознавање на текст 15

3.2 Оптичко препознавање на знаци 18

3.2.1 Обработка на слика 18

3.2.2 Препознавање на примероци 19

3.2.3 Користење на контекстот во процесот на препознавање 21

4 Градба на OCR систем 23

4.1 Структура на типичен OCR систем 23

4.1.1 Оптички скенери 23

4.1.2 OCR хардвер и софтвер 25

4.1.3 Излезен интерфејс 25

4.2 Дигитализација на слика 26

4.2.1 Одмерување 26

4.2.2 Квантизација 27

4.3 Обработка на слика 28

4.4 Анализа на документи 31

4.4.1 Содржина и физичка и логичка структура на документ 31

4.4.2 Генеричка структура на документ 32

4.4.3 Генеричка физичка структура на документи 33

4.4.4 Генеричка логичка структура на документи 34

4.4.5 Декомпозиција и структурна анализа на документи 35

4.4.6 Сегментација на документи 37

4.4.7 Сегментирање на знаци 46

4.5 Препознавање на знаци 57

4.5.1 Препознавање на примероци 58

4.5.2 Препознавање на печатени знаци 61

4.6 Контекстно постпроцесирање 75

5 Систем за препознавање печатени македонски текстови 79

5.1 Корекција на закосеност 81

5.1.1 Проценка на закосеноста на текстот 82

5.1.2 Корекција на закосеност 85

5.2 Сегментација 87

5.2.1 Сегментација на блокови со текст 87
5.2.2 Сегментација на редови со текст 89
5.2.3 Сегментација на зборови 92
5.2.4 Сегментација на знаци 93
5.3 Предкласификација на знаците според формата 96
5.4 Препознавање на изолирани знаци 100
5.4.1 Реализација на системот за препознавање печатени знаци 101
5.4.2 Обучување на класификаторите 109
5.5 Контекстно постпроцесирање 116
5.5.1 Комбинирање на резултатите на класификацијата со употреба на контекстот на ниво на збор 117
5.5.2 Реализација на лексикон за македонскиот јазик 121

6 Заклучок 125

Прилог: Адаптивни логички мрежи 127

Литература 133

1 Вовед

Во досегашниот развиток на човековото општество чувањето и размената на информациите меѓу луѓето се вршело воглавно во вид на пишани документи. Брзиот развојот на науката и техниката во последниве децении, услови појава на нови видови медиуми (пред сé електронски) за чување и размена на информации, меѓутоа и понатаму главен медиум остана хартијата. Масовната компјутеризација која навлезе во сите пори на животот во последниве години предизвика во подготовката и обработката на информациите сé поголема улога да добиваат компјутерите.

Додека информациите печатени на хартија се сосема прифатливи за човекот, кој ваквите информации може лесно да ги прегледува, класификува и обработува, тие се скоро сосема неупотребливи за компјутерот. Компјутерот можеби и многу поточно и побргу би ги обработил или класификувал истите информации, но за тоа е потребно тие да бидат претставени во соодветна, за него разбирлива форма на некој мемориски медиум од кој ќе може да ги прочита. Пренесувањето на печатените информации од хартија во дигитална, за компјутерот разбирлива форма е лесно изводливо со помош на скенер. Меѓутоа, скенерот од аналогниот документ отпечатен на хартија, во меморијата на компјутерот пренесува дигитализирана слика на она што е отпечатено на хартијата, во облик на правоаголна матрица од црни и бели точки (пиксели[1]). Со ова самата информација која можеби е текстуална (низа од букви организирани во наслови, поднаслови, колони, редови, …) или некој технички цртеж (составен од прави линии со различни дебелини, котни линии, шрафури, …) останува скриена во правоаголната матрица од црни и бели точки.

Препознавањето е едно од основните својства на човекот. Додека човекот сосема лесно црните и белите полиња на хартијата ги интерпретира како букви, линии, форми, за компјутерот тие и понатаму остануваат само низи од единици и нули во неговата меморија. Екстрахирањето на основните информации од низата на единици и нули во компјутерската меморија и нивно претставување во форма соодветна за понатамошна обработка со компјутер се сведува на проблемот на препознавање на облици. Облиците кои треба да се препознаат можат да бидат печатени букви, ракопис, технички цртежи, мапи итн.

Автоматското разбирање на документи (Document Understanding) го проучува процесот на екстрахирање на информациите испечатени на хартија и нивна соодветна репрезентација. Еден од клучните чекори во овој процес е оптичкото препознавање на знаци.

---

[1] Pix+El(ement) најмала единица која ја формира сликата.

Оптичкото препознавање на знаци (Optical Character Recognition – OCR) претставува процес на конверзија на скенирана слика со печатен (или пишуван) текст (букви, броеви, специјални знаци или други симболи) во компјутерски обработлива форма (на пример ASCII).

Во рамките на овој труд е развиен систем за оптичко препознавање на печатен македонски кириличен текст. Системот е наменет за помош на лица со оштетен вид и е дел од поголем систем кој вклучува и претворање на текст во говор и печатење со браево писмо. Системот вклучува сегментирање на печатени текстови, одредување на редоследот на читање, изолирање на знаците и препознавање на изолираните знаци. Изолираните знаци се распоредуваат во неколку групи врз основа на нивниот облик, а класификацијата се изведува со тежинско гласање на адаптивни логички мрежи. Конечното препознавање се изведува со користење на контекстот на ниво на изолирани зборови. За потребите на контекстната анализа беше реализиран и компјутерски лексикон за македонскиот јазик.

Организацијата на трудот е следна:

Во поглавјето 2 (OCR систем) се анализирани писмата, елаборирана е потребата од автоматското препознавање на текстови и даден е кус преглед на историјатот на развојот на оптичкото препознавање на знаци. На крајот е претставен системот за помош на лица со оштетен вид.

Поглавјето 3 (Препознавање на документи) го разгледува глобалниот проблем на препознавањето на документи и научните области кои се применуваат во неговото решавање.

Во поглавјето 4 (Градба на OCR систем) е претставена структурата на типичен систем за оптичко препознавање на знаци, варијациите и различните пристапи за решавање на одредени проблеми кои се среќаваат во литературата.

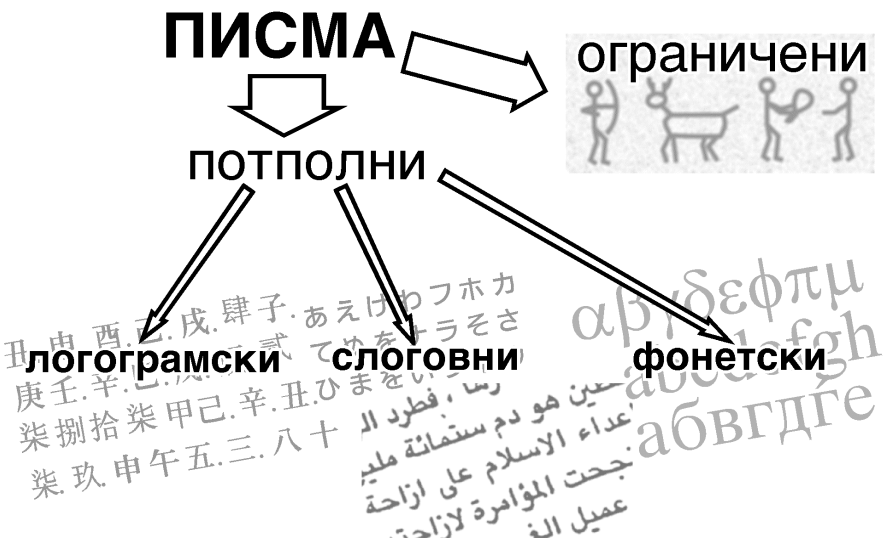
Во поглавјето 5 (Систем за препознавање печатени македонски текстови) е претставена структурата, реализацијата и функционирањето на системот за препознавање македонски текстови, а во поглавјето 6 е даден заклучокот.

2 OCR системи

2.1 Историјат и поделба на писмата

Иако говорот е поприроден начин за комуницирање меѓу луѓето, се смета дека развојот на културата и цивилизацијата се должи токму на развојот на писмото. Писмото претставува метод за комуникација меѓу луѓето со употреба на знаци кои формираат систем. Основни карактеристики на писмото се:

- се состои од вештачки знаци нанесени на површина;
- неговата цел е да пренесе некоја информација;
- оваа цел се постигнува со особината на знаците на одреден начин да го претставуваат говорот.



Слика 2.1 Поделба на писмата

Системите за пишување (писмата) може да се поделат на ограничени и потполни системи (Слика 2.1). Кај ограничените системи има голема мерка на неодреденост бидејќи кај нив не постои директна кореспонденција помеѓу пишаните знаци и говорот. Поради ова интерпретирањето на ваквите системи за пишување е обично независно од јазикот. Потполните системи за пишување се способни да го изразат секој концепт кој може да биде формулиран преку говорниот јазик, и кај нив постои повеќе или помалку цврста кореспонденција меѓу знаците на писмото и елементите на говорот кој го претставуваат.

Елементите на говорниот јазик можат да бидат зборови, слогови или фонеми (гласови – најмали единици во јазикот кои претставуваат различни искажувања). Според ова и писмата може да се поделат на зборовни или логограмски, слоговни и фонетски или алфабетски.

Логограмските писма се карактеризираат со огромен број на знаци од кои секој претставува еден збор или дури и неколку поврзани зборови. Кај слоговните писма, постои знак за секој слог кој се јавува во јазикот, и тие имаат помалку различни знаци (ретко повеќе од 200). Кај фонетските писма постои знак за секој глас (фонем) кој се јавува во јазикот и има најмалку различни знаци. Повеќето од писмата кои постоеле и постојат се комбинирани.

Се смета дека најстарото писмо се појавило околу 3000 години пред нашата ера кај Сумеријанците во Месопотамија, а стотина години подоцна се појавиле и египетските хиероглифи. Во текот на историјата, развојот на писмата постепено се движел од логограмските кон фонетските. Така на пример кинеското писмо користи идеограми, значи во основа е логограмско како и писмата кај повеќето источни јазици. Кај јапонскиот јазик пак, во употреба се две писма од кои едното е идеограмско (канџи), а другото слоговно (хирагана/катакана). Најголем број од останатите писма кои денес се во употреба во светот, се комбинација меѓу фонетското и слоговното писмо (арапски), или скоро чисто фонетски писма како кај повеќето европски јазици. Македонскиот јазик има чисто фонетско писмо.

Кај фонетските писма азбуките се составени од примитивни елементи – знаци кои се користат за претставување на гласовите. Секое писмо има сопствено множество на знаци, познати како букви кои се карактеризираат со својата форма. Секое писмо има и правила за комбинирање на буквите во повисоки јазични форми – зборови. Освен буквите, секое писмо има и ознаки и правила за претставување на броевите, а вклучува и некои специјални знаци кои се користат за означување на интерпункцијата.

2.2 Потреба од OCR

Во почетокот записите биле длабени на глинени плочи, камен или дрво, а подоцна биле пишувани на папирус и хартија. Со појавата на првата печатница во XV век се овозможила репродукција на пишаните документи, во печатена форма. Кај ваквите документи текстот е претставен како визуелна слика на вискоконтрастна позадина, каде формите на знаците припаѓаат на фамилии на фонтови.

Од неодамна стана возможно да се чуваат и прегледуваат електронски подготвени документи на компјутерските екрани. Со ова практично стана можно документите (поточно информациите содржани во нив) доколку се во електронска форма лесно и брзо да се пренесат на далечина и да бидат широко достапни. Хартиените документи кои се аналоген медиум, може да се конвертираат во дигитална форма со процесот на скенирање и дигитализација од кој како резултат се добива дигитална слика. Зависно од потребите (деталите на документот, меморискиот капацитет предвиден за негово сместување или расположливиот пропусен опсег за негова трансмисија) се одредува резолуцијата

при скенирањето. На пример со скенирање на една А4 страница со резолуција од 300 dpi[1] (околу 12 точки по милиметар) во 256 нијанси на сиво се добива слика од преку 8Mb, односно од 1Mb ако се скенира како црно-бела слика.

Од друга страна, една А4 страна која содржи само текст (зависно од големината на буквите) собира 2000–5000 букви кои доколку се кодираат како текст зафаќаат соодветно 2–5Kb. Заштедата во мемориски простор би била повеќекратна ако документот се чува во неговата текстуална форма од која во секое време може повторно да се добие во печатена форма, а не директно неговата слика. Друг, можеби уште поважен факт е дека над текстуалната форма на документот може да се вршат различни обработки кои не се можни доколку тој е достапен само како слика. Корисникот најчесто го интересира информацијата содржана во документот, а поретко неговиот изглед, меѓутоа доколку е потребен и изгледот, документот може да се кодира така, да освен текстот содржи и информации со помош на кои може да се добие во изворната форма со истите употребени фонтови, големини на букви, распоред на текстот на страницата, па дури и графика.

Со големиот технички напредок и масовната компјутеризација во сите пори на животот, голем дел од документите уште во процесот на нивното создавање се расположливи во дигитална форма, но сепак, главен медиум за нивно чување и трансфер и понатаму е хартијата. Исто така, постојат и голем број на нови, а особено постари документи кои не се расположливи во дигитална форма. Поради обработка, архивирање и едноставно пребарување понекогаш е потребно ваквите документи да бидат пренесени во дигитална форма.

Дигитилизираните документи дури и сместени само како слика, зафаќаат многу помалку простор (физичката големина на дигиталниот медиум) отколку пишаните документи и полесни се за одржување (не бараат строго контролирани климатски услови). Меѓутоа скоро истите перформанси за чување на документи, освен лесната преносливост на далечина ги има и микрофилмот. Вистинската предност, дигитализацијата на документите ја добива доколку тие се чуваат во нивната текстуална форма. Во ваква форма над нив може да се вршат најразлични автоматски обработки. Може лесно да се врши пребарување низ нив и за неколку секунди нивната содржина да стане видлива на компјутерскиот екран, а по потреба и да се испечатат на хартија. Од друга страна документот во аналогна форма (на хартија) е само еден и може истовремено да биде достапен само на еден корисник, додека во дигитална форма тој може да биде достапен истовремено до повеќе корисници.

Поради ова уште со појавата на првите сметачки машини била воочена потребата од систем кој ќе може од документите веќе испечатени на хартија да ги екстрахира текстуалните информации сместени на нив. Првиот чекор: добивањето на дигитална репрезентација на документот во вид на слика во меморијата на компјутерот била релативно лесно остварена. Меѓутоа препознавањето на формите (буквите) присутни на оваа слика, кое е сосема едноставна и рутинска работа за човекот се покажало како прилично тежок проблем за компјутерот.

[1] dpi (dots per inch) или ppi (pixels per inch) – точки по инч

2.3 Историјат и поделба на OCR

Корените на автоматското препознавање на знаци датираат уште од далечната 1870 година кога Carey го пронашол ретина-скенерот, кој претставувал систем за трансмисија на слика со помош на мозаик од фотоќелии. Првата примена му била за помош на визуелно хендикепирани луѓе, а првиот успешен обид бил направен од страна на рускиот научник Тюрин во 1900 година. Првите модерни оптички препознавачи на знаци се појавуваат по 1940 година со појавата на првите дигитални компјутери.

Во почетокот пречукувањето било единствен начин информациите од печатена форма да се пренесат во компјутерот. Уште од почетокот на развојот на компјутерската техника биле правени обиди оваа рутинска, монотона и временски долготрајна работа да биде автоматизирана, поточно со што е можно помала човечка интеракција да се овозможи пренесувањето на информациите од хартија во компјутерот.

Во 1949 година италијанскиот Језуит[1] Roberto Busa дошол во претставништето на IBM во Њујорк со многу храбра и напредна (за тогашните прилики) замисла: создавање на електронски индекс на делата на Tomas Aquinas[2]. Гледајќи наназад од денешна гледна точка мора да се признае дека тој создал вистински споменик во полето на примената на компјутерите во културата: 6 милиони дупчени картички, 80.000 страни на компјутерски генериран текст и од неодамна импресивен CD-ROM.

Уште на почетокот на Busa му било јасно дека за да ја оствари својата замисла ќе му биде потребна некаква техника за процесирање на сите информации. Така и завршил кај IBM по подолга обиколка на универзитетите во САД. Кога му било одговорено дека IBM-овите машини не се во состојба да го направат она што тој го бара, тој го извадил од sидот на чекалната постерот со мотото усвоено од IBM на кој пишувало: “The difficult we do right away; the impossible takes a little longer” – (“Тешкото го решаваме веднаш; за невозможното ни треба малку време”) и му го покажал на тогашниот директор на IBM Tomas Votson Sr. и додал: “Не е во ред да се откажете уште пред да се обидете”. Се разбира Busa ја добил поддршката на IBM, иако немал воопшто пари.

За среќа, од тогаш до сега, времињата се промениле и техниката доста напредувала, па неговата работа денес е сосема вообичаено скоро потполно да и се препушти на техниката. Се смета дека модерната OCR технологија е родена во 1951 година со пронајдокот на M. Sheppard, GISMO – робот кој можел да чита и пишува. Во 1954 година J. Rainbow развил прототип на машина која била во состојба да чита печатени големи букви со “фантастична” брзина од една буква во минута. Неколку компании меѓу кои и IBM, Recognition Equipment, Inc., Farrington, Control Data и Optical Scanning Corporation, се појавиле на пазарот со свои производи за препознавање печатен текст до 1967 година. Во доцните шеесети години оваа технологија доживеала многу драматични промени, но OCR системите сеуште се сметале за егзотични и футуристички, и биле користени

[1] Член на здружението на Исус, црковен ред основан од Свети Игнатиус од Лојола во 1534 година.

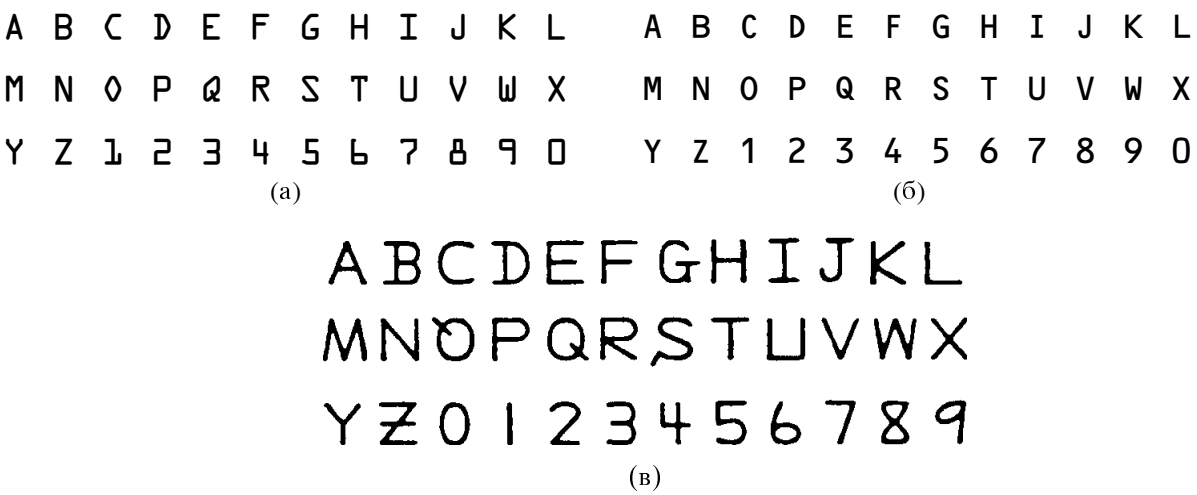
[2] Saint Thomas Aquinas (1225-1274), италијански калуѓер од Доминиканскиот ред, теолог и филозоф. Изразит пример на Сколастицизмот кој прв ги применил Аристотелските методи во Христијанската теологија. Негово најпознато дело е Summa Theologica (1266-1273).

само од страна на владините организации или големите корпорации. Сосема нормално било ваквите системи да чинат над милион долари.

Во почетните години на развојот на OCR системите биле развиени повеќе стандарди за да го олеснат и подобрат оптичкото препознавање на знаците. Некои од овие стандарди се:

- множество на знаци специјално конструирани за да се олесни нивното автоматско оптичко препознавање (OCR-A), ANSI X3.17-18;
- друго множество на знаци исто така специјално конструирани за да се олесни нивното автоматско оптичко препознавање (OCR-B), ANSI X3.49-75;
- хартија за употреба во OCR системите, ANSI X3.62-87;
- мастило за печатење на знаците наменети за OCR, ANSI X3.86-80;
- позиција на знаците наменети за OCR, ANSI X3.93-81.

Формите на буквите од двете множества на ANSI стандардни знаци наменети за оптичко препознавање, како и множеството на стандардни со рака напишани печатни букви се прикажани на Слика 2.2.



Слика 2.2 Стандардизирани фонтови (а) OCR-A, (б) OCR-B и (в) пишувани печатни букви

Денес OCR системите се многу поевтини, побрзи и поточни. На пазарот постојат OCR програми за персонални компјутери кои чинат неколку стотини долари, и препознаваат по неколку стотини знаци во минута, со точност од над 99% кај квалитетно печатените документи [JKN93]. Некои од нив се рекламираат како *омнифонт* односно способност да препознаат било кој печатен фонт, а некои од нив обезбедуваат и можност за обучување со цел да се подобри препознавањето на знаците кои ги користи корисникот или дополнителни знаци.

Иако 99% звучи како фантастична точност, таа се постигнува само во идеални случаи кај квалитетно печатените контрастни документи со не премногу мали букви, па дури и тогаш тоа значи една грешка на секои стотина знаци, што се сведува на по една грешка на секои два реда текст или 10–20 грешки по страна текст. Кај неквалитетно печатените материјали, материјали печатени на неквалитетна хартија (дневен печат), изобличени во процесот на умножување

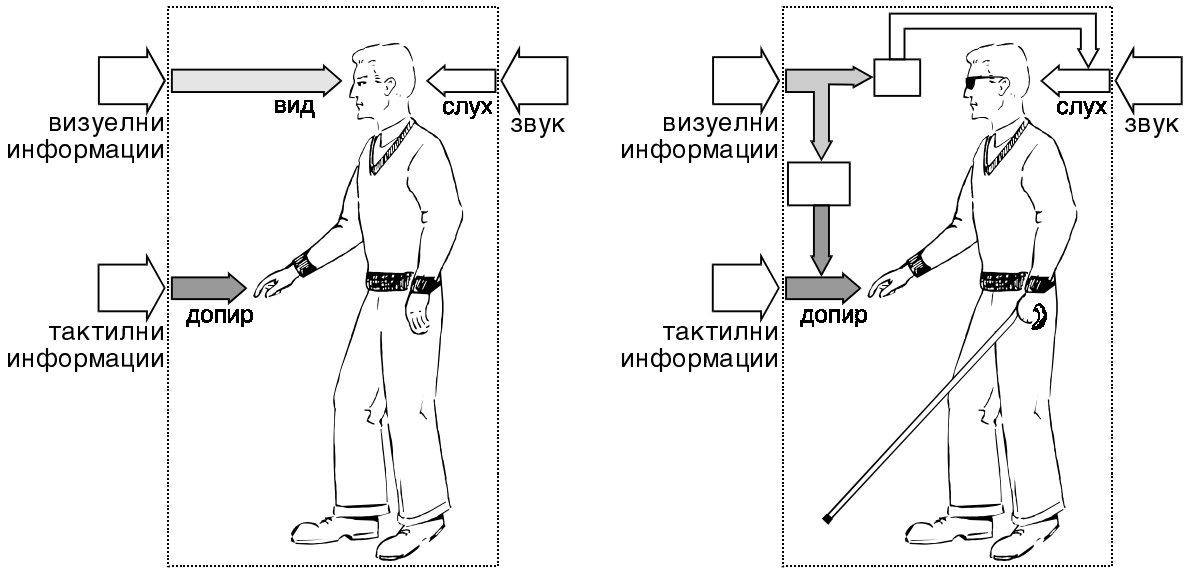
(фотокопија), со премногу мала резолуција (телефакс) или со многу мали букви, точноста при препознавањето е уште помала [RKN93, RKN94, RKN95].

Од 1950 година наваму развиени се стотици OCR системи а многу од нив се и комерцијално достапни. Комерцијалните OCR системи можат да се поделат воглавно на две групи: OCR системи за општа намена и OCR системи за специјална намена (task-specific). OCR системите за специјална намена работат со специфични (строго дефинирани) типови на документи. Некои најчести примени на OCR системите за специјална намена се: читањето на банкарските чекови, адресите на поштенските пратки, слипови од кредитни картички, и др. OCR системите за општа намена се дизајнирани така да можат да препознаваат поширока лепеза на документи како деловни писма, технички извештаи, страници од книги, весници и др. Од аспект на излезот, кај OCR системите за специјална намена обично се поставуваат построги барања од аспект на точноста и брзината или давање на излезни податоци погодни за некоја специфична обработка, и тие обично се дизајнирани да работат без или со што помала интеракција на корисникот. Кај голем број вакви системи кај кои точноста е примарна, непрепознатите и сомнителни документи обично се проследуваат на мануелна обработка од страна на човек. Кај OCR системи за општа намена големо внимание се посветува и на што поверната репродукција на препознатиот документ, зачувување на распоредот на текстот и сликите на страницата и големината и обликот на буквите (употребените фонтови) кои потоа се достапни во форма во која можат да бидат понатаму обработувани. Голем број од нив ја нудат можноста за интерактивно поправање на грешките во препознавањето од страна на корисникот и мануелно внесување на податоците кои не успеале успешно да ги препознаат, а некои од нив овие информации ги користат за интерактивно обучување, односно прилагодување кон податоците кои ги препознаваат.

Покрај постојаното подобрување на точноста и брзината на системите за препознавање на печатени текстови, во последно време сé повеќе се работи во областа на препознавање на ракописни текстови, како и во читањето на обрасци (документи чија структура е строго дефинирана). Користењето на контекстот за подобрување на препознавањето што се употребува и при препознавањето на печатените документи, доби неодменливо значење кај препознавањето на формулари и ракописи. Соодветно и интересот на истражувачите во последно време повеќе се насочи кон истражувањата во областа на користењето на контекстот при препознавањето.

2.4 Компјутерски систем за помош на слепи лица

Луѓето ги користат своите сетила како видот, слухот, допирот, мирисот и вкусот како информациони влезови за примање на информации од околината, или за меѓусебна размена на информации (Слика 2.3). Доколку некое од сетилата (информационите влезови) престане да функционира, можеби е можно дел од информациите наменети за ова сетило да се пренасочат кон некои од другите сетила откако ќе бидат обработени и доведени во соодветна форма во која би можеле да бидат презентирани на другите сетила.



Слика 2.3 Информациони влезови на човекот како систем

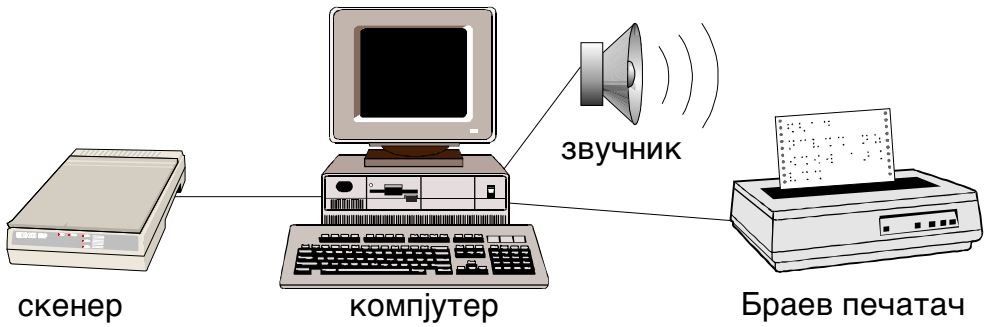
Лицата со оштетен вид, во недостаток или намалени можности да примаат информации со сетилото за вид, се обучуваат овој недостаток да го надокнадат со примање на што повеќе информации со останатите сетила, пред сé со сетилата за допир и слух. Тие не се во можност да ги примаат графичките информации било да се тие прикажани на екран, било да се отпечатени на хартија. За да ги примат информациите тие треба да им бидат презентирани како звучни информации – говор или како браево писмо[1] на специјална хартија или браев терминал од каде можат да ги примат со сетилото за допир. За таа цел печатените текстови треба да бидат или прочитани на глас и при тоа снимани на магнетофонска лента за понатамошна репродукција на лицата со оштетен вид, или отчукани на специјална хартија со помош на специјална машина за браево писмо, односно внесени во компјутерот преку тастатура, подготвени за печатење и потоа испечатени на браев печатач. И едниот и другиот начин вклучува ангажирање на лице кое гледа, кое треба да го прочита на глас текстот пред микрофон или да го внесе во компјутер преку тастатура, а ако е потребно и едното и другото и негово двојно ангажирање. Се разбира ваквиот издавачки фонд е многу ограничен и не секогаш актуелен.

Во светот постојат повеќе различни системи и уреди за помош на визуелно хендикепирани лица [GP93]. Типично се состојат од скенер или камера и модули за OCR и претворање на текстот во говор. Меѓутоа, секој од нив е прилагоден за точно одреден јазик. За македонскиот јазик ваков уред не постои и тоа ниту од аспект на препознавањето на печатен кириличен текст на македонски јазик, ниту од аспект на претворањето од текст во говор.

Системот за препознавање на печатен кириличен текст на македонски јазик е дел од поголем систем за помош на луѓето со оштетен вид. Тој е во развој, како заеднички проект на Електротехничкиот факултет и Центарот за рехабилитација на деца и младинци со оштетен вид “Димитар Влахов” од Скопје

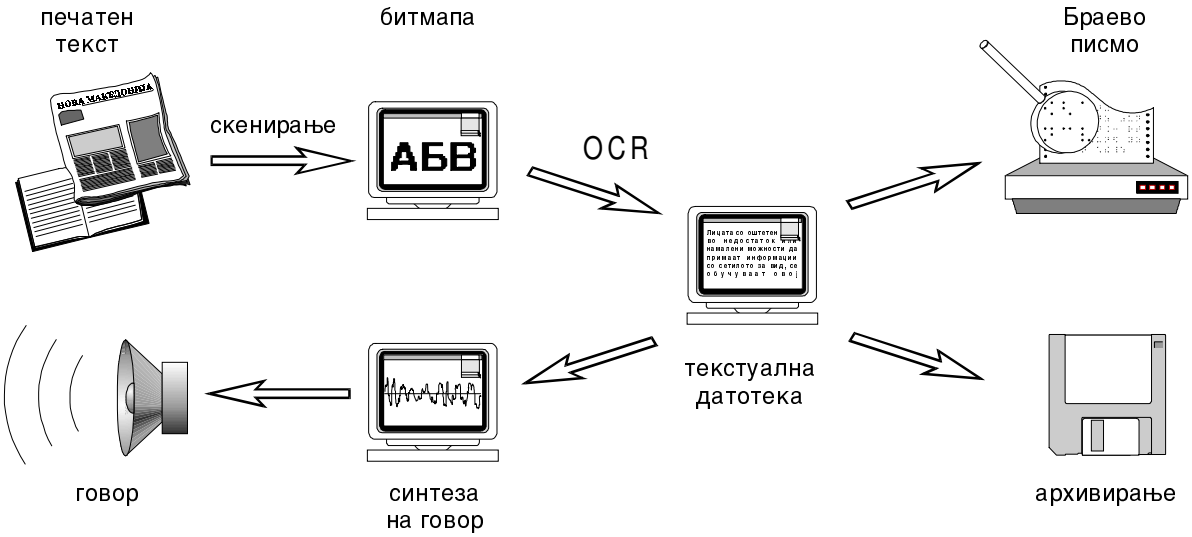
[1] Релјефно писмо кај кое различниот распоред на издигнати точки репрезентира буква или цифра, која слабовидите и слепи луѓе можат да ја одредат со допир. Прв пат бил воведен во 1829 година од страна на францускиот музичар и негов изумител Louis Braille (1809-1852) кој самиот останал слеп на возраст од три години.

[MGK93]. Се очекува со овој систем донекаде да се ублажи постоечката состојба во издаваштвото за слепи и слабовиди лица, бидејќи тој ќе овозможи автоматско читање на печатени македонски кирилични текстови и нивно печатење на браев печатач или изговарање со синтетизатор на говор. На овој начин ќе им се овозможи на лицата со оштетен вид, како следење на дневниот печат и актуелната литература, така и нејзино размножување.



Слика 2.4 Систем за помош на луѓе со оштетен вид

Идејата на системот за помош на луѓе со оштетен вид е автоматизирање на оваа операција со помош на сметач. Системот се состои од персонален сметач на кој е поврзан графички скенер, браев печатач и синтетизатор на говор (Слика 2.4). Печатените материјали со помош на графичкиот скенер се снимаат и се пренесуваат во компјутерот во вид на (црно-бела) слика. Оваа слика понатаму се проследува на обработка на модулот за оптичко препознавање на букви, кој како резултат дава текстуална датотека, која понатаму може да се проследи на печатење на браевиот печатач, да се архивира на некој дигитален медиум или да се проследи на понатамошна обработка на модулот за синтеза на говор (Слика 2.5).



Слика 2.5 Препознавање на текст, печатење, архивирање и синтеза на говор

Од аспект на оваа поделба OCR подсистемот за помош на слепи лица е некаде помеѓу OCR системите за општа намена и OCR системите за специјална

намена. Тој треба да задоволи и некои специфични барања диктирани од хендикепираноста на корисниците за кои е наменет:

- да се овозможи прифаќање на што е можно поширока лепеза на различни документи;
- системот треба да е потполно автономен, т.е. не смее да се надева на било каква помош од корисникот во процесот на препознавање;
- податоците за изгледот на документот се од второстепено значење.

Што се однесува до прифаќањето на што е можно поширока лепеза на различни документи системот е сличен на OCR системите за општа намена, со таа разлика што податоците за изгледот на документот (распоредот на текстот на страницата, големината и типот на употребените букви) се од помало значење. Имено тие се потребни само за одредување на неговата логичка структура поради одредување на редоследот на читање, додека физичката структура не е од интерес. Што се однесува до автономноста, системот е сличен на OCR системите за специјална намена со таа разлика што кај овие системи е дозволено отфрлање на документите кои не се препознати со соодветна сигурност и нивно проследување на мануелна обработка. Кај OCR системот за помош на слепи лица оваа опција не е на располагање, па системот ќе **мора** во секој случај да донесе некаква одлука и да се потруди тоа да биде оптималната според она што му стои на располагање.

3 Препознавање на документи

3.1 Автоматска интерпретација на документи

Задачата на автоматското препознавање на документи е нивната содржина од хартија да се претстави кодира во соодветна електронска форма. Ова кодирање може да биде во различна форма и на различно ниво на апстракција:

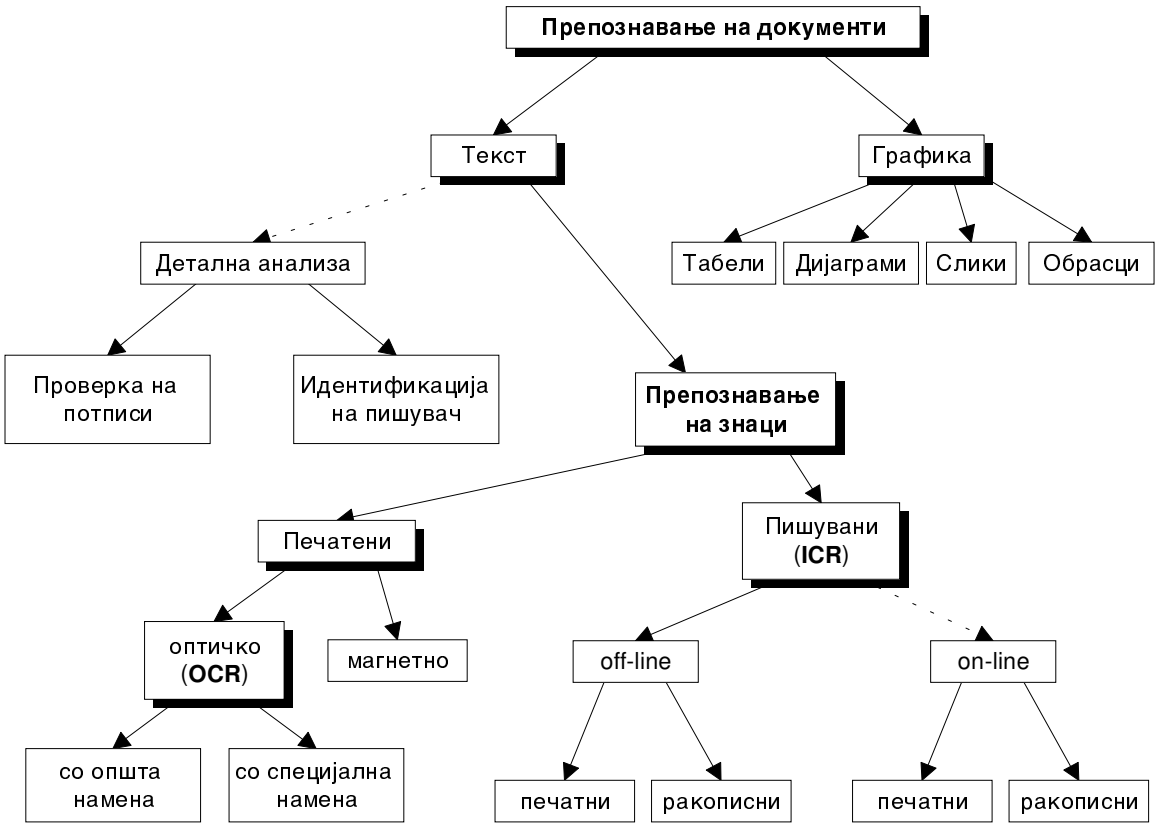
- форма која би овозможила едитирање (интервенции) на документот;
- концизна репрезентација од која документот би можел да биде точно реконструиран (да се добие идентична копија на оригиналот) или
- семантички опис на информациите содржани во него кои може да се употреби за одговарање на прашања (query).

Пред да се интерпретираат, документите мора да поминат низ процесот на нивно препознавање. Одвојувањето на корисните информации од скенираниот документ бара развој и интеграција на повеќе подсистеми. Структурната поделба на препознавањето на документи е прикажано на Слика 3.1. До денес не постои систем за комплетно препознавање на потполно произволно форматирани документи. Зависно од областа на примена и деталноста на пристапот, одредени подсистеми може да бидат понагласени додека други да бидат и потполно отсутни. Глобално, препознавањето на документи може да се подели на препознавање на текст и препознавање на графика.

3.1.1 Препознавање на графика

Препознавањето на графика ги обработува деловите од документот кои не се чисто текстуални, а тука влегуваат: табелите, дијаграмите, сликите и обрасците. Обработката на обрасци би можела да се издвои и како посебна област поради својата специфичност и широк спектар на комплексност. Специфично за системите за обработка на обрасци е тоа што скоро секогаш се специјално изведени за да работат со документи чија форма обично е строго дефинирана, често и специјално дизајнирана за да овозможи што поедноставна електронска обработка [GD96]. Наједноставните случаи на обработка на обрасци (form processing) опфаќаат само проверка на присуство или отсуство на специјални ознаки или облици на точно одредени места на документот. Таква е на пример обработката на ливчињата од лото или спортска прогноза, како и одговорите од некои тестови како општообразовниот дел од приемниот испит на факултетите. Покомплицираните случаи на обработка на обрасци опфаќаат обработка на

различни (дури и рачно пополнувани) формулари, а вклучуваат препознавање на печатени и ракописни текстови. Карактеристично за нив е што обично максимално ги експлоатираат контекстните познавања за подобрување или верификација на препознавањето.



Слика 3.1 Структурна поделба на препознавањето на документи

Табелите, дијаграмите и сликите се често интегрални компоненти на некои документи. Тие ја имаат карактеристиката на дијаграмска репрезентација на информациите. Нивната интерпретација е во врска со можноста на човековиот мозок за аналогно, односно просторно резонирање. Тие се користат за експлицитна претстава на информациите кои инаку би можеле да се претстават само имплицитно доколку се користи некоја друга репрезентација.

Табелите во документите се користат како ефикасно средство за истовремено претставување на информации, својства или особини карактеристични за неколку ентитети. При тоа нивната меѓусебна положба ја диктира релацијата меѓу нив. Препознавањето на табелите е слично на препознавањето на обрасците и вклучува издвојување на вертикалните и хоризонталните линии и препознавање на текст.

Дијаграмите се користат за претставување на информациите кои човекот полесно ги разбира кога се претставени визуелно. Во оваа категорија спаѓаат мапите, техничките цртежи, блок дијаграмите и др. Зависно од целта на нивното препознавање која може да биде само трансформација на нивната репрезентација од печатена во дигитална форма поради архивирање и понатамошна лесна репродукција или екстрахирање на информациите содржани во нив заради нивна понатамошна независна употреба, понатамошната обработка може да биде доста различна. Обработката на дијаграмите во основа

вклучува конверзија на растерската во векторска репрезентација т.е. конверзија од бинарната претстава на линиите како низи од пиксели во соодветна претстава на множество од поврзани јазли и сегменти. Сегментите обично претставуваат примитиви како прави линии, параметарски криви, специфични симболи и текст. Соодветно деловите од сликата кои содржат текст мора да се претстават во соодветна кодирана репрезентација (ASCII) додека графичките информации треба да бидат соодветно препознати и кодирани на друг начин. На линиските сегменти од своја страна треба да им се придружат параметри како: почетна позиција, должина, ориентација, дебелина, тип и др. Визуелната обработка на дијаграмите опфаќа: одвојување на текстот од сликата што обично не е тривијален проблем и негово проследување кон модулот за препознавање на текст, векторизација на остатокот од цртежот, препознавање и обработка на специфични графички симболи како и некои специфични обработки зависни од типот на дијаграмот [SH+92]. Обработката на дијаграмите обично се реализира на повеќе хиерархиски поставени нивоа. На најниското ниво се екстрахираат контурите (графичките информации) и текстот. На следното ниво се пронаоѓаат примитивите како полигони, кругови и агли, а на највиското ниво се врши интерпретација на примитивите за да се извлече концептуално разбирање на дијаграмот. Овие информации потоа се кодираат соодветно на целната репрезентација како команди во некој графички јазик, CAD/CAM датотека или некоја друга претстава [BC+92, VT92].

Препознавањето на слики веќе потпаѓа во доменот на компјутерската визија (computer vision) и може да се разгледува како посебна област. Нејзини специфични примени се обработката на сателитски фотографии, радарски и сонарни снимања, тродимензионална репрезентација на објекти фотографирани од различни агли, роботска визија и низа други специфични примени. Сликите кои се вклучени во документите обично носат дополнителни информации кои се во контекстот на информациите содржани во текстот на документот. Кај препознавањето на документи, сликите обично не се подложуваат на подетална обработка од одредување на нивните димензии и положба во документот.

3.1.2 Препознавање на текст

Препознавањето на текст може да се подели на препознавање на знаци и на детална анализа. При деталната анализа обично не се врши директно препознавање на знаците со кои е напишан текстот, туку се екстрахираат некои други параметри од сликата на текстот, кои се посоодветни за понатамошна обработка. Во оваа област спаѓаат проверката на потписи и идентификацијата на пишувач. Проверката на потписи и детекцијата на фалсификувани потписи го обработува проблемот на прифаќање или отфрлање на хипотезата дека даден потпис припаѓа на даден автор. Проблемот на идентификација на пишувач е доста поширок и тој вклучува одредување дали даден примерок на ракопис припаѓа на некој од поголем број пишувачи.

Деталната анализа, строго гледано и не спаѓа во областа на препознавање на документи, туку само во областа на препознавање на текстови (бидејќи обработува слики или поинакви информации кои во суштината претставуваат текст) и тоа само делумно бидејќи за да ја постигне својата цел не врши експлицитно препознавање на суштинските информации содржани во текстот.

Од друга страна деталната анализа, иако врши анализа на слика со текстуални информации, по начинот на обработка е многу слична на обработките карактеристични за препознавањето на графика.

Препознавањето на знаци може да се подели на препознавање на печатени знаци односно печатени текстови (machine-generated) и препознавање на пишувани знаци односно ракописи (hand-generated). Препознавањето на печатените текстови од аспект на начинот на кој податоците се пренесени од хартија во компјутерот се дели на оптичко и магнетно. Без разлика дали документот бил оптички скениран или бил печатен со магнетно мастило и провлечен под магнетен читач, методите за понатамошна обработка и препознавање се слични и во двата случаја и се опфатени во областа на интерес на оптичкото препознавање на знаци (Optical Character Recognition – OCR) иако епитетот оптичко не одговара за документите кои се магнетно прочитани. Во пракса оптичкото препознавање на знаци во себе често вклучува и делови од автоматската анализа на документи, а некои автори и целата област на препознавање на знаци (било печатни било ракописни) ја означуваат како OCR. Од своја страна од аспект на намената системите за оптичко препознавање на знаци може да се поделат уште и на системи за општа намена и системи за специјална намена.

Областа која го истражува препознавањето на (со рака) пишуваните текстови се нарекува и *интелигентно препознавање на знаци* (intelligent character recognition – ICR). Полето на интелигентното препознавање на знаци од аспект на начинот на кој се собрани податоците може да се подели на on-line и off-line препознавање, од кои секое препознавањето на поврзаните ракописни и одвоените печатни со рака пишувани букви ги разгледува посебно. On-line препознавање (уште наречено и реалновременско или динамичко препознавање) значи дека знаците се препознаваат додека корисникот ги пишува (со евентуално мало задоцнување поради процесирањето). Off-line препознавањето пак се изведува откако пишувањето е завршено. Строго гледано, on-line препознавањето не спаѓа во областа на препознавањето на документи бидејќи вклучува обработка на динамички информации кои кај печатените документи не постојат, но спаѓа во областа на препознавање на текстови.

Off-line препознавањето обично вклучува анализа на слика добиена со скенирање на претходно напишан текст. Иако може да се направат извесни претпоставки за писмото, видот (поврзан ракопис или одвоени букви) и распоредот на текстот на документот, генерално гледано проблемот на off-line препознавањето на ракописи има многу од карактеристиките на класичното препознавање на печатен текст. Рачно напишаните знаци се сепак доста покомплексни од печатените поради многу поголемите дозволени варијации во стилот. Поради ова сегментирањето (изолирањето) на рачно напишаните знакови е доста покомплексно отколку кај печатените текстови. Комерцијалните системи кои препознаваат ракописни знакови често го ограничуваат доменот само на големи букви, изолирани знаци, знаци со специфична форма, ограничен речник и др. Проблемот на препознавање на поврзан ракопис е уште покомплексен, поради фундаменталниот проблем на сегментирањето на буквите, кој без употреба на контекстот и паралелно препознавање и сегментација е неизводлив. Затоа при препознавањето на континуален ракопис класичниот пристап на сегментација на знаците проследен со нивно препознавање не е

прифатлив, па најчесто се користат методи за симултано препознавање и сегментација.

Врвните системи за off-line препознавање на изолирани пишувани букви, во идеални услови (контрастна неизобличена слика, читливо напишан текст со букви кои не се допираат и не се премногу мали и со употреба на речник) достигаат точност на препознавањето од околу 90%. Off-line препознавање на ракописи е област во која последниве години интензивно се работи. Лабораториските системи со одредени ограничувања (само еден корисник, читко пишување и присуство на релативно ограничен речник) постигаат резултати од 40% до 80%.

On-line препознавањето се карактеризира со употреба на динамички информации за препознавање на симболите кои се добиваат при движењето на покажувачот (stylus) по специјална површина за пишување (tablet). Во стриктен смисол on-line препознавањето ги игнорира трагите кои ги остава зад себе инструментот за пишување при неговото движење. Тоа ги зема во предвид (за обработка) само информациите во вид на x-y координати на покажувачот како функција од времето и евентуално некои слични параметри како притисокот врз подлогата, аголот на моливот во однос на нормалата на површината за пишување, брзината на движењето и забрзувањето.

Таблите за дигитализирање (tablets) првпат се воведени во педесеттите години, а по конструкцијата можат да бидат: електростатски, осетливи на притисок, акустички, ласерски, оптички, и др. Типичните табли за пишување имаат резолуција од 200 точки по инч, фреквенција на семплирање од 100 точки во секунда и индикација на "пишување" односно подигање и спуштање на моливот на површината. Сигналот што се добива од нив може да се разгледува како еднодимензионален сигнал во функција од времето. Обработката на ваквиот сигнал опфаќа елиминирање на несаканите варијации како паузите и одвишните точки, линеарна апроксимација и интерполација меѓу точките, нормализација на закосеноста, големината и должината на потезите. Екстракцијата на особините и класификацијата обично се базираат на некоја од класичните техники.

Додека при препознавањето на изолирани (одвоени една од друга) букви, границата меѓу буквите сосема лесно се пронаоѓа, кај on-line препознавањето на континуален ракопис одвојувањето на буквите е нетривијален проблем, за чие решавање се користат слични методи како и кај off-line препознавањето на рекопис. Некои од овие методи при одлучувањето го користат и контекстот и тоа на ниво на потези, парови букви или цели зборови. Денешните комерцијални системи за on-line препознавање на изолирани букви, во идеални услови (само еден корисник, читливи букви, прилагоденост на системот кон корисникот) постигаат точност од околу 90%, а со употреба на лимитиран речник точноста достига скоро 100%. Најдобрите резултати во on-line препознавањето на ракопис се нешто послаби.

Предноста на on-line над off-line препознавањето е што тоа располага со динамички информации за самиот процес на пишувањето. Во нив спаѓаат информации како бројот на потези, нивниот редослед, насоката во која е повлечен, како и брзината на испишувањето на секој од потезите, а кај некои табли за дигитализирање и притисокот на површината, и аголот на моливот.

Друга предност на on-line препознавањето е неговата интерактивност, односно тоа најчесто се одвива за време на самото пишување, па во случај на погрешно препознавање или непрепознавање корисникот може веднаш да интервенира, а системот динамички да се прилагодува кон ракописот на корисникот.

3.2 Оптичко препознавање на знаци

Оптичкото препознавање на знаци е една од клучните фази во процесот на препознавање на документи. Од своја страна тоа претставува поле за примена на научните достигнувања од повеќе области како: обработката на слика (image processing), препознавањето на примероци (pattern recognition) и процесирањето на природните јазици (natural language processing – NLP).

Обработката на слика ја опфаќа обработката на документот од неговото скенирање до издвојувањето на сликите на изолираните симболи кои треба да се препознаат. Препознавањето на примероци всушност врши препознавање на облиците и нивна класификација, односно пресликување од доменот на слика (правоаголна матрици со црни и бели точки) во доменот на симболи (букви, цифри, специјални знаци, …). Со употреба на семантичкото знаење од повисоко ниво врз база на контекстот, можни се значителни подобрувања на препознавањето. Бидејќи најчесто се работи за препознавање на текстови на некој јазик, процесирањето на природните јазици исто така наоѓа своја примена при оптичкото препознавање на знаци. Иако на прв поглед се чини дека обработката на слика, препознавањето на примероци и постпроцесирањето со експлоатација на контекстните информации, се три дисјунктни и последователни процеси, често пати тие се така испреплетени што не постои јасна граница меѓу нив.

3.2.1 Обработка на слика

Некои техники за препознавањето на примероци и нивна класификација можат да се применат директно врз расположливите податоци. Меѓутоа кога податоците претставуваат слики или делови од слика, понекогаш подобро е од нив првин да се издвојат некои особини, врз база на кои тие ќе се класифицираат. Пред издвојувањето на овие особини од сликата, корисно е таа на некој начин да се обработи за да се отстранат ирелевантните информации или шумот и да се подобрат нејзините карактеристики кои понатаму би го олесниле издвојувањето на потребните особини од неа.

Ваквата операција може да се нарече **слика-во-слика** (picture-to-picture) трансформација бидејќи оригиналната слика се заменува со нова слика. Во наједноставниот случај оваа трансформација ја следи **слика-во-особина** (picture-to-feature) трансформација која екстрахира мал број на особини од сликата. По неа доаѓа **особина-во-одлука** (feature-to-decision) трансформацијата која конечно ги класифицира објектите на сликата. Во пракса се применуваат повеќе циклуси на обработка на сликата, нејзина анализа и донесување некои одлуки пред да се дојде до конечните резултати.

Обработката на слика (Image processing) опфаќа различни слика-во-слика трансформации, со кои се постигнуваат различни ефекти како: промена на осветленоста и контрастот на сликата, бинаризација (tresholding), филтрирање, одредување на прагот за бинаризацијата, нормализација на рангот, осветленоста и контрастот, заматување, изоструване, детекција на рабови и др. [GJJ96, Dav90]. Во обработка на слика спаѓаат и ротацијата, скалирањето (промена на нејзините димензии), поместувањето со или без интерполација.

Пред преземање на некоја обработка врз сликата треба да се процени која обработка да се примени и со кои параметри. За да се постигне ова се врши **анализа на сликата**. Некои од најчесто применуваните анализи на сликите се: детекција на линии, проценка на закосеност, лоцирање на објектите, сегментирање и бележење на регионите на сликата, следење и репрезентација на границите меѓу регионите, пресметување на проекции, профили и хистограм на осветленост (gray level histogram), идр.

По извршената анализа и соодветна обработка на сликата, се лоцираат областите од интерес и се издвојуваат некои нивни особини, врз база на кои ќе треба да се изврши препознавање и класификација.

3.2.2 Препознавање на примероци

Зборот pattern (примерок, модел, шаблон, калап), потекнува од зборот патрон и во оригиналната употреба означува нешто што претставува совршен пример за имитирање. Препознавањето на примероци (Pattern recognition) го има значењето на идентификација на идеалот според кој даден објект е направен. Во психологијата препознавањето на примероци е дефинирано како процес во кој "екстерните" сигнали кои доаѓаат во сетилата се претвораат во препознатливи перцептуални искуства.

Едно од клучните прашања кое се наметнува е како се создава концептот за идеалното. Тој може да се формира преку дедуктивен или индуктивен процес. Во првиот случај се смета дека концептот за идеалното е вроден кај набљудувачот, додека кај индуктивното создавање на концептот, набљудувачот го создава концептот за идеалното преку набљудување на многу неидеални (несовршени) примероци. Овој процес може да се нарече и учење, и тоа учење со учител доколку примероците се означени како претставители на еден или повеќе идеали, или учење без учител доколку примероците не се означени.

Менталните процеси со кои се постигнува препознавањето се очигледно сложени и сеуште недоволно познати. Препознавањето на примероци не му е својствено само на човекот, тоа е присутно и кај животните и имало значајна улога во еволуциона смисла. При тоа, генерализацијата овозможила организмот успешно да се снајде во нови ситуации, и преземајќи слични акции или однесување, кои се покажале успешни во претходното искуство со ситуации од ист или сличен тип, успешно да излезе од нив.

Нервниот систем на човекот е многу поефикасен во справувањето со проблемите кои се од животно значење за човекот отколку со оние кои не се. И не е за чудење што човекот релативно лесно успева да изведе препознавање на комплексни примероци, но има тешкотии со множењето повеќецифрени броеви. Фактот што човековиот нервен систем не е прилагоден за брзо изведување на

нумерички операции не́ изненадува, имајќи го предвид фактот што способноста да се изведуваат вакви операции и немало некое особено значење за опстанокот на единката (во биолошка смисла на зборот) се́ до минатиот век.

Појавата на дигиталните компјутери стави на располагање машини кои далеку ги надминуваат човековите способности за комплексни нумерички операции. Се постави и прашањето дали овие машини ќе се покажат подеднакво успешни и во ненумеричките операции, особено во менталните функции како препознавањето на примероци. Проблемите од оваа категорија генерално беа адресирани како *вештачка интелигенција* или машинска интелигенција. Се покажа дека раниот оптимизам во поглед на способноста на компјутерите да решаваат проблеми од класата на машинската интелигенција не беше оправдан. Човековиот мозок има многу повисок степен на комплексна организација и од најнапредните компјутери.

Сепак со развојот на науката границите на она што компјутерот не може да го направи полека но сигурно се поместуваат. Постојат многу работи во чија автоматизација е потребно препознавање на примероци. Сортирањето на поштата и внесување на податоци во компјутер бараат препознавање на печатени или ракописни букви. Броењето на крвните зрнца, проверката на исправност и контролата на квалитет, како и автоматизираното составување на механички делови генерално бараат препознавање на облици. Слични барања има и во автоматската анализа на сателитски слики, радарски сигнали, подводни сонарни снимања и др.

Во пракса под препознавањето на примероци се подразбира автоматска детекција и класификација на објекти или настани. Мерките и податоците кои се користат за класифицирање на објектите се нарекуваат *особини* (features), а прототиповите или категориите во кои тие се распоредуваат се нарекуваат *класи*. Постојат техники (како кластерирањето на пр.) кои овозможуваат ненадгледувано учење (учење без учител) односно автоматско откривање на класите, кои се обидуваат да го поделат множеството податоци во природно појавни групи без предетерминирана структура на класите. Во пракса најчесто постои множество класирани податоци кое може да се употреби при дизајнирањето на автоматскиот систем. Индивидуалните единици, објекти или настани кои треба да се класираат се нарекуваат **примероци** (или мостри). Множеството класирани примероци кое се употребува во дизајнот на системот се нарекува **множество за обука** (training set), додека множеството исто така класирани примероци кое се употебува за тестирање, евалуација и нагодување на некои дополнителни параметри на системот се вика **множество за тестирање** (испитување) (test set, validation set).

Понекогаш класата на објектот не може да биде одредена по некој апсолутен критериум, туку зависи од мислењата на повеќе експерти. Во вакви случаи податоците треба да бидат независно класифицирани од страна на неколку експерти, а потоа нивните резултати да бидат некако комбинирани. Дури и ако некој од експертите понекогаш греши во класирањето, мнозинското гласање или некој друг систем на гласање во групата веројатно ќе доведе до точен резултат. Комбинирањето на повеќе експерти во одлучувањето исто така носи и дополнителна информација за доверливоста (level of confidence) на одлуката донесена од групата. Ако сите експерти се сложиле во заклучокот, доверливоста на ваквата одлука е голема, ако пак мислењата биле поделени,

доверливоста ќе зависи од распределбата на гласовите односно дисјункноста на нивните мислења.

3.2.3 Користење на контекстот во процесот на препознавање

Контекстот, односно околината во која се наоѓа примерокот кој треба да се препознае, често пати значително го ограничува изборот на можните класи на кои тој може да припаѓа. Користењето на овој факт при препознавањето, може значително да ја подобри точноста на препознавањето.

Најуспешен во препознавањето на печатени или ракописни текстови и понатаму е човекот. Постојат непобитни докази за тоа дека процесот на читање не е само едноставно препознавање на букви. Повеќе студии од кои некои датираат уште од XIX век, покажуваат дека човекот полесно ги препознава буквите како дел од збор, отколку како изолирани [REL94]. Зборовите од своја страна претставуваат комбинација од повеќе букви, меѓутоа сите можни комбинации од одреден број букви секако не претставуваат валидни зборови. Така на пример од шест букви можат да произлезат $31^6 = 887.503.681$ "зборови", но само мал дел од нив – неколку илјади, односно помалку од 0.001% претставуваат валидни зборови. За да се искористи овој факт, читателот треба да го познава јазикот на кој е напишан текстот, односно сите или барем поголемиот дел од зборовите кои се во оптек. Затоа, читањето на текстови напишани со позната азбука но за читателот непознат јазик, значи во отсуство на контекст, поточно за читателот непознат контекст, е многу потешко. Истражувањата на движењата на окото за време на процесот на читање покажуваат дека при вообичаеното читање само околу 68% од зборовите се фиксираат, сугестирајќи дека познавањето на контекстот на повисоко ниво придонесува во процесирањето на преостанатите 32% [REJ94], што значи дека не секој збор може да се најде на било кое место во реченицата.

Аналогно, и кај системите за автоматско препознавање на текстови со експлоатација на познавањето на контекстот можат значително да се подобрат перформансите. Со вклучување на речник (лексикон) во системот за препознавање, се овозможува користење на контекстните информации на ниво на зборот. Користењето на овие дополнителни информации при препознавањето доведуваат до намалување на неодреденоста и подобрување на перформансите на системот. Освен лексичките, можно е и вклучување на контекстни информации од повисоко ниво во процесот на препознавањето, како што се синтаксата и семантиката на јазикот. Описот на синтаксата и семантиката на природните јазици не е едноставна и таа е област на истражување на процесирањето на природните јазици.

Употреба на контекстот за подобрување на препознавањето е присутна и кај обработката на податоци кои не спаѓаат во природните јазици. Такво е на пример препознавањето на адресите на поштенските пратки и употреба на поштенскиот број за препознавање на останатиот дел од адресата (пред сé името на местото и евентуално на улицата), или обратно [SSR94]. Слично кај препознавањето на банкарски чекови или уплатници, износот напишан со цифри се проверува со препознавањето на истиот напишан со зборови [Sri96]. Контекстот може значително да го подобри препознавањето на некои специфични податоци како листи на потези од шаховски партии, каде со

семантичка анализа на можните потези во дадена ситуација резултатите на препознавањето се подобрени од 40% на 98% [BT90]. Слично, искористување на лексичкиот и семантичкиот контекст би можело да се примени и за подобрување на препознавањето на скенирани листинзи на компјутерски програми.

4 Градба на OCR систем

4.1 Структура на типичен OCR систем

Типичниот OCR систем се состои од три логички компоненти (Слика 4.1):

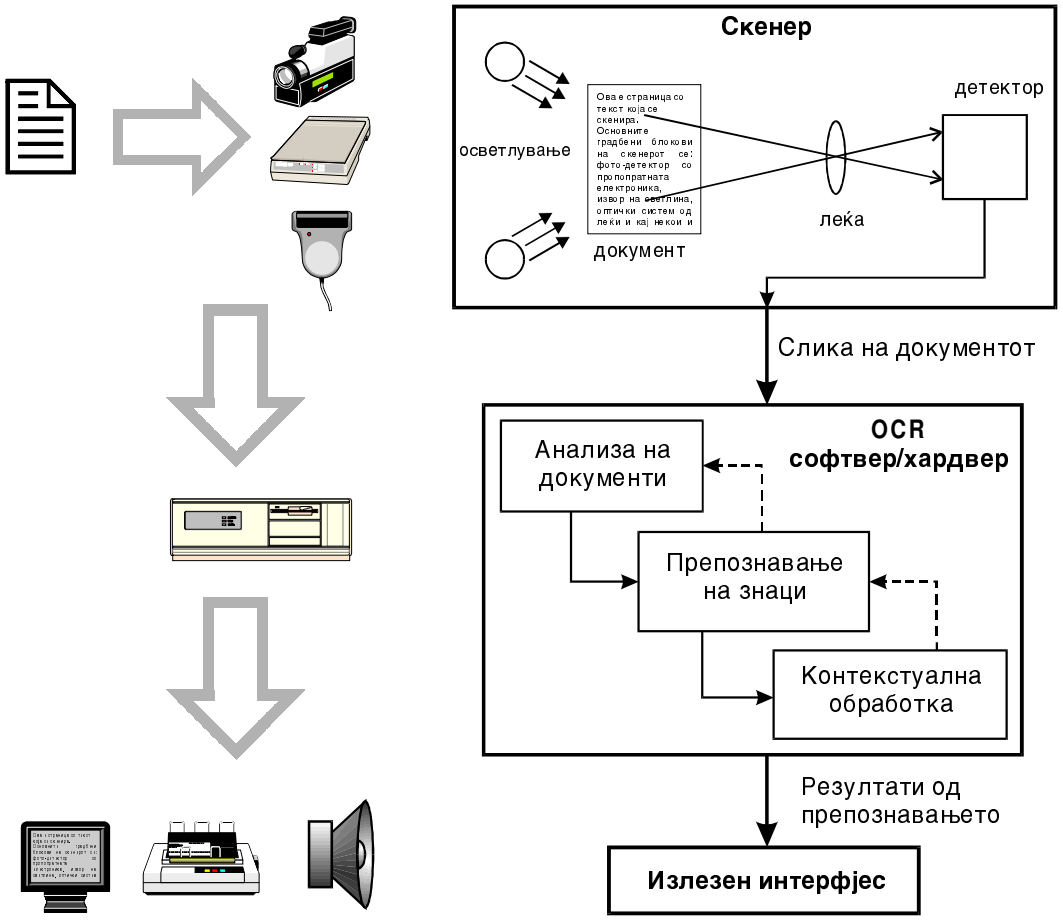
1. Уред за прифаќање на дигитализираната слика на текстот (скенер);
2. Хардвер и софтвер за OCR;
3. Излезен интерфејс.

Уредот за прифаќање на сликата на текстот кој треба да се препознае најчесто е оптички скенер, меѓутоа може да биде и видео камера, или друг систем за трансмисија на слика (телефакс на пр.). Сликата на текстот се обработува со помош на хардверот и софтверот за OCR. Овој процес обично опфаќа три операции: анализа на документот (екстракција на индивидуални слики на поедините букви), препознавање на буквите (врз основа на нивната форма) и контекстно постпроцесирање (за исправање на грешките при препознавањето или за ограничување на изборот при препознавањето). Зависно од реализацијата на системот, овие три операции може да бидат интегрирани меѓу себе така да границите меѓу нив не се видливи. Излезниот интерфејс е одговорен за комуникација на OCR системот со надворешниот свет, било да е тоа непосредниот корисник или друг систем кој ја продолжува обработката на препознатиот текст.

4.1.1 Оптички скенери

Основните градбени блокови на скенерот се: фото-детектор со пропратната електроника, извор на светлина, оптички систем од леќи а некаде и систем за транспорт на документи. Системот за транспорт го поставува документот во просторот за скенирање, изворот на светлина ја осветлува површината на документот, и преку системот од леќи сликата на документот се проектира на фото-детекторот. Детекторот се состои од матрица на фото осетливи елементи од кои секој светлината која паѓа на него ја претвора во аналоген електричен сигнал пропорционален на јачината на светлината која паѓа на него. Овие аналогни сигнали потоа се обработуваат и се пренесуваат во компјутерот. Скенирањето се одвива со помош на детекторот и меѓусебното движење на детекторот и/или документот.

На пазарот постојат повеќе типови на скенери кои може да се поделат според техничката изведба, прецизноста, брзината, максималниот формат кој може да го примат, максималниот број на поддржани бои или нијанси на сиво и др. Професионалните скенери се во состојба да скенираат по неколку стотини страни со A4 формат во минута со резолуција од 400 ppi. Тие најчесто се црно-бели или поддржуваат само нијанси на сиво (gray-level) и секогаш се со вграден систем за транспорт на документи, а чинат по неколку десетици илјади долари.



Слика 4.1 Структура на OCR систем

Кај полупрофесионалните скенери отсуствува уредот за автоматско транспортирање на документи (кај некои се нуди како опција). Овие уреди уште се наречени и скенери на страници (page scanners). Кај нив документот рачно се поставува од страна на корисникот на стаклената хоризонтална површина за скенирање. Овие скенери иако се доста побавни од професионалните, скенираат цела страна одеднаш без дополнителна интервенција од корисникот. Скенерите на страници чинат околу илјада долари и овозможуваат скенирање на документи со максимална големина нешто поголема од A4 форматот со максимална резолуција од 300 до 600 ppi. Некои од нив овозможуваат и скенирање во боја, а зависно до избраната резолуција и број на бои за скенирање на една A4 страна им е потребно од дваесетина секунди до неколку минути. Рачните скенери (hand scanners) се најевтини и чинат неколку стотини долари. Кај нив за да се оствари скенирањето потребно е корисникот со приближно константна брзина да го движи уредот над површината на документот поставен на рамна подлога. Најчесто овозможуваат скенирање на формати до A5 со максимална резолуција

од 400 ppi, некои дури и во боја. Во последниве неколку години на пазарот се појавија и скенери кои (и според цените и според можностите) се некаде помеѓу рачните и скенерите на страници. Тоа се портабилни уреди (некои произведувачи дури ги вградуваат во тастатурите) кои овозможуваат вметнување на документот во скенерот кој понатаму самиот го транспортира низ себе со помош на вртлив цилиндер слично како кај телефакс уредите. Недостатокот на овие уреди е што не можат да скенираат материјали отпечатени на тврди подлоги (картон) како и поврзани материјали (книги, часописи) ниту пак документи поголеми или многу помали од A4 форматот.

4.1.2 OCR хардвер и софтвер

Хардверско-софтверскиот систем кој го врши препознавањето на буквите од сликата на текстот може да се подели на три чекори: анализа на документот, препознавање на знаците и контекстно постпроцесирање.

За да можат да бидат препознати, неопходно е индивидуалните знаци да бидат изолирани од сликата на останатиот текст. Процесот на лоцирање и одвојување на индивидуалните знаци од сликата на документот е познат како анализа на документот. Сигурното сегментирање и препознавање на знаците многу зависи и од квалитетот на оригиналниот документ и од квалитетот на неговата скенирана слика. Процесите кои се обидуваат да го компензираат лошиот квалитет на оригиналниот документ и/или неквалитетното скенирање вклучуваат обработки за подобрување на квалитетот на сликата (image enhancement) и одстранување на шумот (noise removal).

Двете основни компоненти при препознавањето на знаците се: одвојувањето на особини (карактеристики) (feature extraction) и препознавањето (класификација). Анализата на особини ги одредува карактеристиките или множеството на особини кои се користат за опис на сите знаци. За дадена слика на буква се екстрахираат особините кои таа ги поседува и се проследуваат на класификаторот. Особините и класификаторите кои се користат во OCR системите ќе бидат разгледани подетално во поглавјето 4.5.

Контекстуалните информации може доста корисно да се искористат за подобрување на препознавањето. Нивното наједноставно користење е на ниво на збор, каде проверката на резултатите од препознавањето се изведува преку проверка на постоењето на препознатиот зборот во лексиконот од зборови. Во некои случаи каде класификаторот не донесува цврсти одлуки за припадност на знакот во одредена класа, туку одредува неспецифицирана (fuzzy) припадност кон класите во вид на повеќе кандидати со придружени веројатности, контекстуалните информации можат да се користат и во процесот на финално одлучување за припадност во одредена класа на знакот.

4.1.3 Излезен интерфејс

Излезниот интерфејс овозможува резултатите од препознавањето да бидат префрлени во соодветна форма зависно од областа во која ќе се користат, на некој мемориски медиум или директно во друга апликација. Така на пример тие може да бидат префрлени во текст процесор, база на податоци, табела за

пресметување (spread sheet), синтетизатор на говор и др. Кај системите за специјална намена каде однапред се знае каде и како ќе бидат применети резултатите и барањата што се поставуваат пред нив, системот за препознавање може соодветно да се оптимизира за потенцирање на некои и занемарување на други особини при препознавањето. Така на пример за текст процесорот е особено важно да се пренесат што е можно повеќе информации за распоредот на текстот, големината и стилот на буквите, додека овие информации за синтетизаторот на говор се скоро потполно небитни.

4.2 Дигитализација на слика

Првиот чекор при препознавањето на знаци е секако пренесувањето на сликата на печатениот текстот од аналогна форма во соодветна дигитална форма разбирлива за компјутерот. Дигитализацијата на слика го претставува процесот на конверзија на континуалното поле на сликата во еквивалентен дигитален облик. Таа опфаќа два процеса: *одмерување* и *квантизација*.

За потребите на препознавање на документи, сликата може да се разгледува како статичка (непроменлива во текот на времето) и може да се ограничиме на црно-бели слики, односно да ги занемариме боите. Ваквата слика може да се претстави како дводимензионална функција која е ограничена, реална и ненегативна.

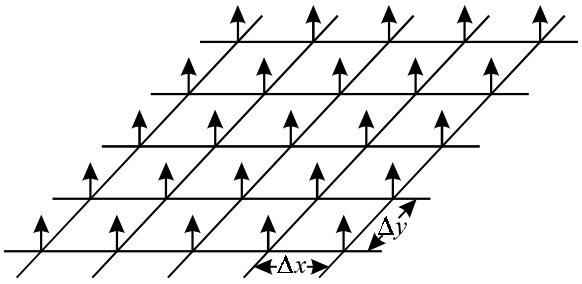
$$0 \le f(x, y) \le C_{\max} \tag{4.1}$$

Секоја ваква функција можеме да ја претставиме со нејзините вредности одбрани во соодветни точки од просторот (рамнината на сликата). Овој процес на избирање на одбироци од континуалниот сигнал на функцијата во соодветни точки се нарекува просторна дискретизација или одмерување. Следниот чекор при дигитализација на сликата е дискретизација по ниво или квантизација. Во овој процес аналогните вредности на одбироците се заменуваат со конечен број на дискретни вредности, односно амплитудно се квантизираат.

4.2.1 Одмерување

Кај системите за одмерување влезните слики се третираат како детерминистички полиња. Нека $f_i(x, y)$ претставува континуално поле на идеална слика со бесконечни димензии. Одмероците на идеалната слика во просторот се добиваат со множење на $f_i(x, y)$ со просторната функција за одбирање која се состои од еден бесконечен низ од Dirac-ови делта функции кои образуваат една просторна решетка (Слика 4.2). Просторната функција за одбирање е:

$$s(x, y) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} \delta(x - i\Delta x, y - j\Delta y) \tag{4.2}$$



Слика 4.2 Низа Dirac-ови делата функции

Во тој случај одмерената слика може да се претстави со релацијата [Boj89]:

$$f_0(x, y) = f_i(x, y)s(x, y) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} f_i(i\Delta x, j\Delta y)\delta(x - i\Delta x, y - j\Delta y) \tag{4.3}$$

Се гледа дека $f_i(x, y)$ се внесува под знакот на сумата што значи дека нејзината вредност се проценува само во точките на одбирање $(i\Delta x, j\Delta y)$. Вредностите Δx и Δy ги претставуваат растојанијата помеѓу две соседни делта функции по x и y оската соодветно и тие директно влијаат на резолуцијата на дигитализираната слика. Овие вредности треба да бидат соодветно избрани за да го задоволуваат Nyquist-овиот критериум.

4.2.2 Квантизација

Функцијата на квантизерот е моменталната вредност на побудниот сигнал, кој во општ случај припаѓа на непребројиво множество на точки, од континуалниот амплитуден опсег да го трансформира во најблиската дозволена вредност од конечното дискретно множество на амплитуди. Се разбира при квантизацијата секогаш доаѓа до грешка, т.н. шум на квантизација. Нека x и $\hat{x}$ се амплитудата на реалниот одбирок на сигналот и неговата квантизирана вредност. При тоа разликата меѓу нив ја претставува грешката при квантизација за тој одбирок. Нека x е случаен одбирок кој се наоѓа во опсегот

$$a_D \le x < a_G \tag{4.4}$$

каде a_D и a_G ги претставуваат долната и горната граница на набљудуваниот опсег соодветно. Проблемот на квантизацијата наложува одредување на низ на нивоа на одлучување d_j и низ на дискретни вредности r_j, така да ако е

$$d_j \le x < d_{j+1} \tag{4.5}$$

одбирокот се квантизира на вредност $\hat{x} = r_j$. Нивоата на одлучување и соодветните дискретните вредности треба да бидат така одбрани за да ја минимизираат грешката при квантизацијата. Квантизерите во основа можат да бидат линеарни (со линеарно еквидистантно распоредени нивоа на одлучување) или нелинеарни.

При дигитализација на слики се квантизира секој одбирок од сликата. Секое ниво на осветленост се кодира со одреден код со зададена должина. Бројот на расположливи нивоа за квантизација зависи од должината на кодот L кој обично е цел степен од бројот два:

$$L = 2^n \quad \text{каде} \quad n \in \mathrm{N} \tag{4.6}$$

Во обработката на документи најчесто се употребува двонивовска квантизација ($n = 1$) т.е. се работи со црно-бели слики. Нешто поретко се употребува линеарна квантизација со 16 или 256 нивоа на сиво, бидејќи ресурсите (и мемориски и процесорски) за обработка на вакви слики се значително поголеми. Најголемиот дел од досегашните истражувања во анализата на документи и оптичкото препознавање на знаци се вршени со црно-бела претстава на документот, па често дури и ако уредот за скенирање дава слика во нијанси на сиво, таа уште на почетокот се сведува на црно бела заради заштеда на меморија и сите обработки понатаму се вршат врз неа.

4.3 Обработка на слика

Обработката на слика (Image Processing) претставува низа на слика-во-слика трансформации со цел да се подобри нејзиниот квалитет или да се отстранат нерелевантните информации од неа. По одмерувањето и квантизацијата сликата може да се претстави како целобројна матрица во која секој елемент претставува една точка од сликата, па обработката на слика се сведува на некакви трансформации над ваквата матрица.

По дефиниција писмото претставува некакви високо контрастни симболи нанесен на некоја површина. Соодветно апликациите за оптичко препознавање на знаци обично со обработките кои се преземаат над сликата на документот тежнеат да го подобрат контрастот со што знаците кои треба да се препознаат бидат што поизразени и полесно воочливи односно препознатливи, како и да се отстранат нерелевантните информации и шумот. Постојат и пристапи [SZP93, RSZ+94] во кои особините за препознавање на знаците се екстрахираат директно од сликата во нијанси на сиво.

Најчестите трансформации кои се преземаат на сликата на документот во процесот на нејзината обработка се: подобрување на контрастот, филтрирање и бинаризација (thresholding). Скоро сите трансформации делуваат локално т.е. новата вредност на пикселот (елементот од матрицата) зависи од неговата претходна вредност (пред трансформацијата) и евентуално од вредностите на пикселите кои се во негова близина (најчесто неговите непосредни соседи).

Подобрувањето (зајакнувањето) на контрастот е една од наједноставните операции која може да се изведе над дигитализирана слика во нијанси на сиво. Ако $f_g(x, y)$ е дигитализирана слика кај која нивоата на сиво се кодирани со вредности од 0 (црно) до L (бело), тогаш зајакнувањето на контрастот може да се дефинира како:

$$F_g(x, y) = \max(0, \min(L, f_g(x, y) \cdot \gamma + \beta)); \quad \forall (x,y) \tag{4.7}$$

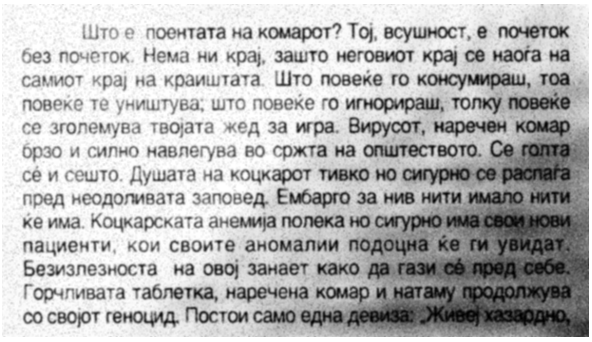
каде $F_g(x, y)$ е новодобиената слика, а γ и β се поволно избрани константи со кои се регулира зајакнувањето на контрастот и глобалната осветленост.

Бинаризацијата (thresholding) е исто така едноставна операција која сликата во нијанси на сиво ја сведува на црно-бела. Тоа се постига со едноставно

поставување на 1 на сите елементи на матрицата чии вредности се поголеми или еднакви од зададен праг, а на 0 на елемнетите со вредности под прагот:

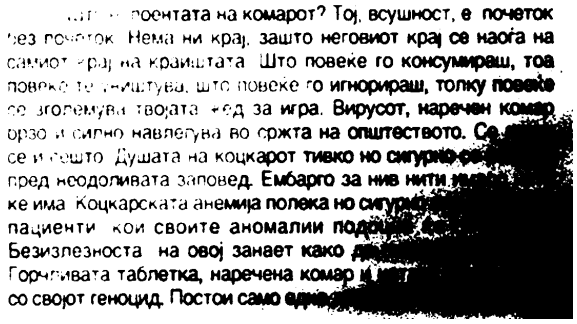
$$F(x,y)=\begin{cases}1, & f_g(x,y)\geq t\\ 0, & f_g(x,y)<t\end{cases};\quad \forall(x,y) \tag{4.8}$$

каде $F(x, y)$ е новодобиената црно-бела слика, а t е прагот. Избирањето на прагот е од особена важност за квалитетот на бинаризацијата и треба да обезбеди минимална загуба на релевантните информации. Пронаоѓањето на оптималниот праг обично повлекува некоја статистичка обработка на сликата во нијанси на сиво пред нејзината бинаризација. Оваа обработка најчесто вклучува пресметување на т.н. *хистограм на интензитети* или распределба на интензитети (intensity distribution) од кој се пресметува оптималниот праг (Слика 4.3 б). Ваквата бинаризација е наречена уште и *униформна бинаризација* бидејќи прагот е со константна вредност за целата слика. Кај слики со нехомогено осветлување (Слика 4.3 а) униформната бинаризација не дава прифатливи резултати (Слика 4.3 в). За ваквите случаи мора да се примени т.н. *адаптивна бинаризација* (Слика 4.3 г) кај која вредноста на прагот се пресметува и применува локално за прозорец со одредени димензии од сликата [Dav90] или за секој пиксел од сликата поодделно. Преглед на 11 локални адаптивни методи и 4 глобални методи за бинаризација се дадени во [TJ95], а оценка на перформансите на некои од методите во [TT95].

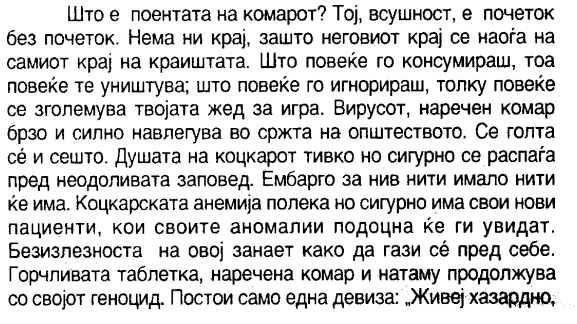

Што е поентата на комарот? Тој, всушност, е почеток без почеток. Нема ни крај, зашто неговиот крај се наоѓа на самиот крај на краиштата. Што повеќе го консумираш, тоа повеќе те уништува; што повеќе го игнорираш, толку повеќе се зголемува твојата жед за игра. Вирусот, наречен комар брзо и силно навлегува во сржта на општеството. Се голта сé и сешто. Душата на коцкарот тивко но сигурно се распаѓа пред неодоливата заповед. Ембарго за нив нити имало нити ќе има. Коцкарската анемија полека но сигурно има свои нови пациенти, кои своите аномалии подоцна ќе ги увидат. Безизлезноста на овој занает како да гази сé пред себе. Горчливата таблетка, наречена комар и натаму продолжува со својот геноцид. Постои само една девиза: „Живеј хазардно,

(а)

(б)


поентата на комарот? Тој, всушност, е почеток без почеток. Нема ни крај, зашто неговиот крај се наоѓа на самиот крај на краиштата. Што повеќе го консумираш, тоа повеќе те уништува; што повеќе го игнорираш, толку повеќе се зголемува твојата жед за игра. Вирусот, наречен комар брзо и силно навлегува во сржта на општеството. Се голта сé и сешто. Душата на коцкарот тивко но сигурно се распаѓа пред неодоливата заповед. Ембарго за нив нити ќе има. Коцкарската анемија полека но сигурно пациенти, кои своите аномалии подоцна Безизлезноста на овој занает како Горчливата таблетка, наречена комар со својот геноцид. Постои само

(в)


Што е поентата на комарот? Тој, всушност, е почеток без почеток. Нема ни крај, зашто неговиот крај се наоѓа на самиот крај на краиштата. Што повеќе го консумираш, тоа повеќе те уништува; што повеќе го игнорираш, толку повеќе се зголемува твојата жед за игра. Вирусот, наречен комар брзо и силно навлегува во сржта на општеството. Се голта сé и сешто. Душата на коцкарот тивко но сигурно се распаѓа пред неодоливата заповед. Ембарго за нив нити имало нити ќе има. Коцкарската анемија полека но сигурно има свои нови пациенти, кои своите аномалии подоцна ќе ги увидат. Безизлезноста на овој занает како да гази сé пред себе. Горчливата таблетка, наречена комар и натаму продолжува со својот геноцид. Постои само една девиза: „Живеј хазардно,

(г)

Слика 4.3 Бинаризација (thresholding): (а) слика во нијанси на сиво, (б) хистограм на интензитети, (в) црно-бела слика добиена со униформна бинаризација, (г) црно-бела слика добиена со адаптивна бинаризација

Филтрирањето е широко применувана метода во голем број инженерски апликации за отстранување на непожелниот шум кое најчесто се имплементира како нископропусен филтер кој работи во фреквентен домен. Кај

дигитализираните слики ваквиот директен пристап не е исплатлив бидејќи добивањето на сликите во фреквентен домен е доста скапа операција. Поцелисходно е за оваа цел да се користи конволуција.

Конволуцијата е моќна и широко применувана техника во обработката на слика и во други области на науката. Конволуцијата на две функции $f(x)$ и $g(x)$ се дефинира како:

$$f(x) \circ g(x) = \int_{-\infty}^{\infty} f(u)g(x-u)du \tag{4.9}$$

Кога конволуцијата се применува на дигитализирани слики горниот израз преминува во:

$$F(x,y) = f(x,y) \circ g(x,y) = \sum_i \sum_j f(i,j)g(x-i, y-j) \tag{4.10}$$

каде g претставува просторна конволуциона маска. Конволуционите маски ретко се поголеми од 5×5, а најчесто се со големина 3×3. Ако со h ја означиме инвертираната маска на g:

$$h(x,y) = g(-x,-y) \tag{4.11}$$

и ако (4.11) ја воведеме во (4.10) се добива поинтуитивна формула за конволуција на дигитализирана слика:

$$F(x,y) = \sum_i \sum_j f(x+i, y+j)h(i,j) \tag{4.12}$$

Во литературата [Dav90] може да се сретнат различни филтри со карактеристични просторни конволуциони маски од кои најупотребувани се:

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \qquad \text{и} \qquad \frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

Лошата страна на филтрирањето со помош на конволуција со конволуциона маска која во основа претставува нископропусен филтер е што предизвикува заматување и изобличување на рабовите (границите меѓу светлите и темните региони на сликата). По цена на нешто покомплексни операции, подобри карактеристики нудат median и mode филтрирањето, но и кај нив е присутно заматувањето на рабовите [Dav90]. Методите за филтрирање кои работат со слики во нијанси на сиво зачувувајќи ги при тоа рабовите се обично итеративни [BC95] и релативно скапи за имплементација. Затоа во OCR апликациите филтрирањето најчесто се применува над веќе бинаризираната слика применувајќи ја методата на т.н. "просејување" (seed filling) [CC95] која се сведува на наоѓање на сите поврзани компоненти (множества на соседни црни точки) и бришење (одстранување од сликата) на оние кои имаат помалку од одреден број елементи. На овој начин ефикасно се отстранува шумот во вид на мали изолирани црни региони во сликата. Границата за отстранување мора да биде соодветно избрана да не предизвика бришење на делови од текстот како некои помали интерпункциски знаци или делови од букви (точката на ј на пр.).

Влијанието на шумот на резултатите од препознавањето на знаци е широко истражувано особено од аспект на апликациите кои вршат препознавање на текстови поради нивно индексирање за натамошно пребарување [TBC+93, TBC+94].

4.4 Анализа на документи

Анализата на документот, поточно анализата на сликата на документот е процес кој врши глобална интерпретација на сликата на документот. Тој треба да одговори на прашањето како сé што е познато за јазикот, писмото, форматирањето на документи, обработката на слика и препознавањето на знаци треба да се искомбинира за да се постигне целта поставена од дадена апликација. Анализата на документи ги обработува глобалните прашања околу препознавањето на печатени (или пишани) документи додавајќи суперструктура над чистото препознавањето на знаци која воведува организација на документот и применува надворешно знаење за да се обезбеди негова комплетна интерпретација.

4.4.1 Содржина и физичка и логичка структура на документ

Информациите содржани во еден документ, независно од нивната организација и физичка претстава ја претставуваат *содржината* на документот. Начинот на кој содржината на документот е организирана на физичкиот медиум ја претставува *физичката структура* на документот. *Логичката структура* пак го претставува начинот на кој е организирана содржината на документот пред тој да биде пренесен на физички медиум и да добие и физичка структура.

Начинот на резонирање при перцепирањето на печатени информации како и природата на писмото наметнува секој документ да има своја логичка структура. Логичката структура на документот претставува хиерархија која ја изразува семантиката на документот. Преку неа се дефинирани значењата на одредени делови од содржината на документот: наслов, автори, апстракт, текст, илустрации, слики и др.

Секој печатен документ има своја геометриска или *физичка структура*. Физичката структура го дефинира распоредот на деловите од текстот и нетекстуалните информации содржани во документот. Помеѓу логичката и физичката структура на документот постои еден спрема многу кореспонденција, односно една иста логичка структура на документ може да биде изразена преку повеќе различни физички структури со едноставна промена на големината и типот на употребеното писмо, распоредот на текстот, растојанието помеѓу пасусите, бројот на колони со текст и др.

Анализата на документи вклучува како откривање на неговата физичка структура, односно геометриско разложување на сликата на документот на нејзините основни физички структурни блокови, така и откривање на семантичкото значење и меѓусебната интеракција на овие блокови односно логичката структура на документот. Процесот на геометриското разложување на

сликата на документот е наречен *геометриска анализа* или разложување, односно *декомпозиција на документот* и тој резултира со множество на блокови кои ја изразуваат геометриската структура на документот. Процесот кој врз база на содржината и меѓусебната положба на блоковите ја изнаоѓа логичката структура на документот е наречен *структурна анализа* на документот.

Одредувањето на структурата на документот обично се реализира врз база на експлицитен или имплицитен модел на класата на документи кои се од интерес. Моделот ја опишува физичката појава и релацијата меѓу ентитетите кои го формираат документот. Иако одредувањето на структурата на документот обично претходи на фазата на препознавање, кај некои пристапи и препознавањето влијае врз одредувањето на структурата, особено логичката. Истражувањата во оваа област може да се поделат воглавно во четири класи од аспект на примената: анализа на текстуални документи, анализа на формулари, анализа на технички цртежи и специјални апликации како анализа на адреси на поштенски пратки и чекови.

Основната цел на анализата на текстуални документи е да се постигне т.н. инверзно форматирање, односно скенираниот печатен документ да се преведе повторно (назад) во јазик за опис на документи од каде ќе може повторно да биде препечатен доколку е потребно. Во некои апликации не е потребна целосна интерпретација на структурата на документот. Така на пример кај системите за препознавање, индексирање и пребарување на технички документи најчесто е доволно да се обработат само насловот, авторите, апстрактот и клучните зборови, додека кај системите за читање (синтеза на говор) на печатени текстови, глобалната анализа на документот е потребна само за одредување на редоследот на читање. Бидејќи најголем број од техниките (пристапите) за препознавање и класификација на знаци се прилагодени да работат со сликите на изолирани знаци (иако во последно време се појавија и некои пристапи кои препознаваат цели зборови), дури и ако логичката структура на документот не е потребна, анализата на сликата на документот е неопходна поради лоцирањето на поединиите знаци од текстот.

4.4.2 Генеричка структура на документ

Логичката структура на документот ја изразува организацијата на содржината на документот, независно од неговата физичка претстава. Претежно текстуалните (text-intensive) документи на пример вообичаено се состојат од реченици, зборови и букви, и евентуално од конструкции на повисоко организационо ниво како пасуси и поглавја кај статиите (написите) или адреса, тело и потпис кај документите за кореспонденција (писмата). Постојат повеќе видови (претежно) текстуални документи (книги, технички документи, писма, меморандуми, написи од весник) од кои секој има специфична логичка структура на повисоките организациони нивоа.

Во обидот да се опишат структурните релации меѓу компонентите на документот, клучно е обезбедувањето на соодветно ниво на апстракција за да се избегне врзување за терминологијата поврзана со одреден тип (класа) на документи. За таа цел се дефинира *генеричка структура* на документ односно генеричка логичка структура и генеричка физичка структура на документ. Таа треба да обезбеди дефинирање на релации меѓу компонентите на документот

независни од типот, како и преминување на *специфична структура* карактеристична за одредена класа на документи односно опишување на специфичните релации меѓу ентитетите карактеристични за соодветната класа документи. Ова е во согласност со концептот на објектно ориентираната анализа (object-oriented analysis – OOA), во која објектите претставуваат инстанци на соодветните класи. Според ова термините како колона текст, ред, збор и симбол во описот на физичката структура или пасус, поглавје и глава во описот на логичката структура на документот се всушност инстанци на генеричките термини дефинирани со генеричката структура. Некои од нив како физичкиот термин "ред" или логичкиот термин "реченица" се применливи кај мнозинството документи, додека некои како логичкиот термин "потпис" се специфични за одредена класа на документи.

4.4.3 Генеричка физичка структура на документи

Генеричкиот модел за опис на физичката структура на документите предложен од Doermann *et al.* [DDS+96] ги дефинира следниве компоненти:

- **Рамка** (frame) – област (регион) од страницата од документот која може да содржи колекција од рамки од пониско ниво и/или блокови.
- **Главна рамка** (root frame) – рамка која го опфаќа целиот физички документ.
- **Страница** (page) – рамка која ја опфаќа целата станица на која документот или негов дел е физички сместен.
- **Множество од страници** (page set) – рамка која опфаќа повеќе страници. Инстанца на множеството страници може да биде на пример поглавје или книга.
- **Блок** или **проста рамка** (block or simple frame) – терминална рамка од најниско ниво која за даденото ниво на деталност не треба или не може да се дели понатаму.

Рамката е дефинирана како рекурзивна компонента која може да содржи множество од една или повеќе рамки. Блокот е дефиниран како терминална рамка која дефинира регион од страницата и има своја содржина. Зависно од нивото на деталност блокот може да претставува ред, збор или знак на пр.

На секоја рамка е придружено множество на атрибути. Информациите за положбата, поравнетоста, димензиите на рамката, и др. се дефинирани со нејзините атрибути. Блоковите, како листови во стеблото на хиерархиската структура на документот се единствените кои имаат своја содржина, додека сложените рамки повисоко во хиерархијата претставуваат листи од покажувачи кон рамките од пониско ниво. Гранулираноста (прецизноста) на блоковите мора да биде доволно фина за да ги опфати најмалите логички компоненти кои треба да се опишат. Така на пример, ако компонентата на најниско ниво во логичката структура е збор, тогаш блокот не може да биде ред, тој мора да биде најмалку збор, но може да биде и знак. Од друга страна пак, на една компонента во логичката структура може да и одговараат две или повеќе компоненти во физичката структура (на пример еден збор да биде разделен во два реда, или еден пасус на две страни). Исто така една физичка компонента може да одговара на повеќе логички компоненти (на пример датумот да биде разделен на ден, месец и

година), но ваквите случаи се поретки бидејќи обично и физичката структура на документот се избира така да ја рефлектира логичката.

Кај текстуалните документи инстанците кои би одговарале на рамките и блоковите од генеричката физичка структура на документот предложени од Doermann *et al.* [DDS+96] се:

- **Знак** (character) – блок кој ја содржи сликата на еден симбол од јазикот на документот.
- **Збор** (word) – рамка која содржи група од еден или повеќе хоризонтално поравнети знаци и може да биде:
  - **Подзбор** (subword) – рамка која содржи група од еден или повеќе хоризонтално поравнети знаци, кои го претставуваат почетокот или крајот на зборот.
  - **Почеток на збор** (preword) – подзбор кој го содржи првиот дел од зборот. Тој е лоциран на крајот на редот и завршува со тире.
  - **Крај на збор** (postword) – подзбор кој го содржи последниот дел од зборот. Тој е лоциран на почетокот на редот и претставува продолжение на почетокот на зборот од претходниот ред.
- **Ред** (line) – рамка која содржи еден или повеќе зборови и најмногу еден почеток на збор и еден крај на збор.
- **Стек** (stack) – рамка која содржи еден или повеќе редови наредени еден над друг.
- **Колона** или столб (cloumn) – рамка која содржи еден или повеќе стекови подредени еден над друг.
- **Страница** (page) – рамка која сорджи една или повеќе колони.

Вака дефинираната структура е директно применлива само кај документите на јазиците кои употребуваат некое од алфабетските писма (латиница, кирилица, алфабета). За документи на кинески јазик на пример чие писмо се состои од идеограми и има поинакви правила за пишување (организација на страницата – од горе - надолу, од лево - кон десно) генеричката структура би била и понатаму применлива, но хиерархијата и инстанците би биле поинакви.

4.4.4 Генеричка логичка структура на документи

Како и физичката структура која е претставена со хиерархија на рамки, и генеричката логичка структура е организирана во форма на стебло чии листови обично се знаците или симболите, а коренот го претставува целиот документ. За да може генерички да се опише логичката хиерархија на документ, а при тоа да не се огрничи на определен број на нивоа и нивни имиња како “поглавие” или “пасус”, Doermann *et al.* [DDS+96] дефинира термин “**текстон**” (texton) како логички аналог на “рамка”. Со ова за опис на генеричката логичка структура на документот се дефинирани следниве компоненти:

- **Текстон** (texton) – логичка компонента кај текстуалните документи која содржи еден или повеќе текстони или прости текстони.
- **Главен текстон** (root texton) – текстон кој го содржи целиот документ, на пример: книга, весник, извештај, писмо и сл.
- **Прост текстон** (simple texton) – логичка компонента кај текстуалните документи кој понатаму не се дели.

- **Сложен текстон** (compound texton) – текстон составен од одвоено заглавие (header), тело (body) и траилер (trailer), на пример песна составена од: заглавие – наслов на песната, тело – една или повеќе строфи и траилер – име на авторот.

Како и рамката и дефиницијата на текстонот е рекурзивна. Рекурзијата завршува со прост текстон кој претставува основна логичка единица, која обично е знакот. Рекурзивната дефиниција на текстонот го опфаќа целиот спектар на логички нивоа во било кој претежно текстуален документ.

Можните инстанци на текстонот кај претежно текстуалните документи се:

- **Знак** (character) – текстон кој претставува симбол во јазикот на документот, кој обично е знак (буква, цифра или специјален знак).
- **Збор** (word) – текстон кој содржи секвенца на еден или повеќе знаци кои имаат некакво значење во јазикот на документот.
- **Фраза** (phrase) – текстон кој претставува препознатлива секвенца од еден или неколку зборови која има некакво значење, а не мора да формира целосна реченица.
- **Реченица** (sentence) – секвенца од една или повеќе фрази која претставува валидна граматички исправна реченица.
- **Параграфон** (paragraphon) – текстон кој претставува генерализација на параграф и се состои од една или повеќе реченици и/или фрази.

Знакот и зборот имаат како логичка така и физичка дефиниција и најчесто имаат директна кореспонденција. Разликата меѓу нив е во тоа што логичкиот знак го претставува самиот симбол, додека физичкиот знак е сликовна претстава на логичкиот знак. Слично, додека логичкиот збор претставува семантички објект, физичкиот збор е или сликовна репрезентација на логичкиот збор или претставува подредена секвенца од физички знаци. На повисоките нивоа директната кореспонденција се губи. На пример, ретко кога една реченица зафаќа точно еден ред.

Освен хиерархиската структура дадена со рекурзивната дефиниција на текстонот, логичката структура треба да го задржи и редоследот на читање на документот. Редоследот на читање (reading order) го специфицира редоследот во кој треба да се проследат знаците или симболите од текстуалниот документ за да бидат правилно разбрани.

4.4.5 Декомпозиција и структурна анализа на документи

Лоцирањето на знаците на сликата на документот скоро секогаш е претходено со декомпозиција и структурна анализа на документот. Декомпозицијата и структурната анализа на документи вклучуваат неколку операции како: одредување на закосеноста на документот и евентуално нејзино отстранување, одвојување на сликите од текстот, раздвојување на текстот на колони, редови, зборови и поврзани компоненти како и нивно соодветно групирање кое претставува погоден геометриски опис на документот.

Геометрискиот опис на сликата на документот претставува хиерархиска спецификација на геометријата (положбата, распоредот) на различни типови на максимални хомогени региони. За еден регион се вели дека е хомоген доколку

целата негова површина е од ист тип, на пример: знак, ред со текст, пасус со текст, дијаграм, слика, … Формално, геометрискиот опис на сликата претставува подредена двојка $\Phi = (\mathcal{R}, \mathcal{S})$, каде $\mathcal{R}$ е множество од региони, а $\mathcal{S}$ е просторна хиерархиска релација над множеството $\mathcal{R}$. Регионот R претставува подредена двојка $R = (T, \theta)$, каде T го дефинира типот на регионот, а θ претставува вектор од параметри кои го определуваат (опишуваат) регионот. Векторот може да вклучува и неодреденост како стандардна девијација или интервал на толеранција за некои од параметрите, претставени како пар. На ниво на целиот вектор пак, неодреденоста може да биде претставена како матрица од ковариjанси.

Методите кои се употребуваат за одвојување на овие региони најчесто го користат фактот дека структурните елементи на документите се обично распоредени во правоаголни блокови поравнети паралелно со хоризонталната и вертикалната оска на страницата. Методите може исто така да користат и дополнителни визуелни, просторни и лингвистички знаења. За одвојување на објектите од позадината на пример, потребно е визуелно знаење. Означувањето на блоковите вклучува употреба на просторно знаење како на пример изгледот (распоредот на информациите) на типичен документ. Препознавањето пак на оштетен текст вклучува употреба и на лингвистичко знаење, како употреба на лексикон од дозволени зборови. Се разбира неопходна е и соодветна интеракција меѓу различните типови на знаење. Доделувањето на логичко значење на одреден текстуален блок на пример, бара употреба на просторно, семантичко и синтактичко знаење како и интерпретација на соседните региони.

Процесот на декомпозиција и структурна анализа на документи може да се подели на три фази:

1. Сегментација на блокови;
2. Класификација на блокови;
3. Логичко групирање на блокови.

Во процесот на сегментација на блокови сликата на документот **се дели** на неколку обично правоаголни региони. Секој блок е хомоген регион кој содржи едно од следново: позадина, текст, слика, дијаграм или табела. За секој блок зависно од типот може да се асоцирани и дополнителни податоци како големината на текстот и стилот за текстуалниот блок, бројот на редици и колони за табелата и сл. Резултат на овој процес е множество на блокови со соодветни својства.

Процесот на класификација, зависно од типот и дополнителните податоци за блокот како и неговата просторна поставеност на документот врши соодветно **означување** на блоковите како: позадина, текст, слика, дијаграм или табела.

Во процесот на логичко групирање и подредување на блоковите се врши **групирање** на блоковите во групи кои имаат значење за читателот (наслов, обичен текст, слика) и имаат пошироко значење отколку едноставната физичка декомпозиција на блокови. Резултатот на ваквото групирање најчесто е некакво хиерархиско стебло од рамки каде структурата е дефинирана со формата на стеблото и информациите сместени во неговите јазли. Ваквото стебло од рамки може да се конвертира во некоја од стандардните претстави за дигитални документи како SGML (Standard Generalized Markup Language) [TCB+94].

4.4.6 Сегментација на документи

Методите кои се користат кај сегментацијата на документи можат глобално да се поделат на *top-down* и *bottom-up* пристапи. Кај top-down пристапот документот се дели на главни региони кои понатаму рекурзивно се делат на подрегиони врз база на структурата на документот. Bottom-up пристапот почнува со екстрахирање на мали компоненти кои потоа се обединуваат во поголеми региони. Top-down методите го градат стеблото на хиерархиската структура почнувајќи од неговиот корен – целиот документ, кој потоа постепено рекурзивно го делат на сé помали и помали региони. Bottom-up методите обработката ја започнуваат на ниво на пиксели кои потоа ги обединуваат во поврзани компоненти, поврзаните компоненти ги обединуваат во букви, буквите во зборови, зборовите во редови итн., односно стеблото на хиерархиската структура го градат од листовите кон коренот. Се среќаваат и хибридни методи кои претставуваат комбинација на top-down и bottom-up пристапот.

Голем број од методите за сегментација се доста чувствителни на закосеноста на сликата на скенираниот документ (бидејќи очекуваат блоковите со текст да бидат правоаголни и паралелни со рабовите на страницата), па соодветно пред сегментацијата со вакви методи мора да се спроведе корекција на закосеноста (skew corection/elimination) на документите. Скоро сите top-down методи се доста чувствителни на закосеност, додека некои од bottom-up методите иако првичната сегментација (на поврзани компоненти и букви) ја вршат на оригиналната (некорегирана) слика на документот, при нивното обединување во поголеми региони (зборови, редови) обично ги користат информациите за закосеноста на документот.

Top-down методите обично ги спроведуваат следниве постапки при сегментација на документите:

1. Корекција на закосеност
2. Сегментација на блокови
3. Класификација на блокови
4. Организација на блоковите

Секој текстуален блок понатаму се дели на редови со текст, а потоа во секој ред се лоцираат зборовите. Сегментирањето на знаците во склоп на зборот е посебен проблем кој посебно се разгледува.

Корекција на закосеност

Закосеноста на документот претставува изобличување на идеалната слика на документот на тој начин што редовите со текст кои вообичаено се паралелни со работ на страницата, не се паралелни со работ на сликата. До појавата на закосеност на сликата на документот доаѓа во процесот на дигитализација на документот поради неидеалноста на операторот или механизмот кој го поставува документот на површината за скенирање. Поради осетливоста на некои од методите за сегментација на закосеноста на документите, често е неопходно да се спроведе корекција на закосеноста.

Корекцијата на закосеност на скенираната слика на документот опфаќа два процеса:

1. Детекција и проценка на закосеноста (skew detection and estimation)
2. Корекција на закосеноста (skew corection/elimination)

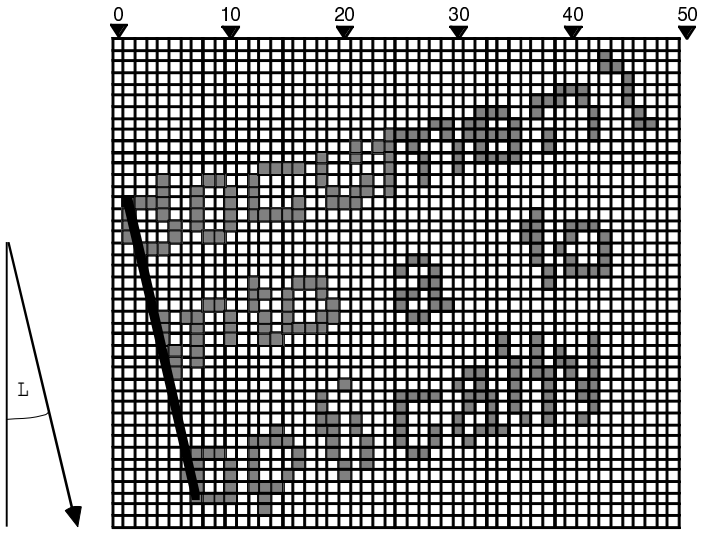
Kaj top-down методите детекцијата и проценката на закосеноста најчесто се врши над самата слика на документот, додека корекцијата најчесто претставува едноставна ротација на сликата за проценетиот агол. Kaj bottom-up методите детекцијата и проценката може да се одвива и врз множеството поврзани компоненти, а проценетиот агол на закосеност да се користи при нивното обединување во поголеми региони и за евентуално индивидуално ротирање на секоја поврзана компонента, без директно ротирање на целата слика.

Слика 4.4 Типичен закосен ред со текст

За агол на закосеност на скенираните документи се смета аголот кој редовите текст го формираат со хоризонталата (Слика 4.4), односно доминантниот агол на мнозинството редови на страницата. Соодветно, повеќето методи проценката на овој агол ја вршат со анализа на текстуалните зони од документот. Прецизноста со која треба да се одреди аголот на закосеност зависи од методата која се користи за сегментација на текстот, како и од самиот документ (пред сé од големината на буквите). Просечен оператор при вообичаена внимателност го поставува документот на површината за скенирање со грешка помала од ±5°. Најголем број на top-down методите за сегментација бараат грешката на закосеноста да е помала од "неколку" степени. Од друга страна некои од проблемите поврзани со сегментацијата и препознавањето на текст можат полесно и поточно да се решат доколку постои поточна проценка на закосеноста на текстот. Така на пример, текстовите со долги густо печатени редови со мали букви (на пр. редови со текст со големина од 6 pt долги 16 cm), кога се проектираат хоризонтално интерферираат веќе кај закосеност од само ⅓° [Bai87]. Прецизното одредување на базната линија и x-висината на редовите со текст, бара уште попрецизно одредување на аголот на закосеност.

Некои пристапи аголот на закосеност го одредуваат со анализа директно на сликата на документот или некоја нејзина трансформација (Houhg или Fourier), додека други вршат анализа на поврзаните компоненти, односно некои од нивните атрибути како локцијата и односот на бели и црни површини во рамките околу поврзаните компоненти.

Dengel [Den90], ја употребува LMS (Left Margine Search) процедурата за одредување на аголот на закосеност. Според оваа постапка се бараат позициите на појавата на првата црна точка во секоја линија од сликата и се сместуваат во вектор. Во следниот чекор врз база на овие точки се бараат прави линиски сегменти кои ја одредуваат левата маргина на текстот или текстуалните блокови (Слика 4.5). Градиентите на овие линиски сегменти скалирани во однос на нивната должина даваат дистрибуција над сите можни агли, чија доминантна вредност (максимум) го одредува аголот на закосеност $\mathfrak{L}$ на документот во однос на вертикалата.



Слика 4.5 Одредување на закосеноста на документот со LMS методата

Liu *et al.* [LLS92] употребуваат слична процедура со тоа што тие ја проценуваат закосеноста на долната маргина користејќи ги поврзаните компоненти на веќе сегментираната слика. Пристапот на Liu *et al.* се состои во пронаоѓање на основните линии на знаците од последните неколку редови со текст. За да се игнорираат знаците кои одат под основната линија (g, j, p, q, y) се врши филтрирање на поврзаните компоненти и отфрлање на оние кои имаат поголема висина од просечната. На крајот аголот на закосеност се проценува како средна вредност од сите агли помеѓу две последователни точки. Претходните два пристапа релативно брзо и со задоволителна точност ја определуваат закосеноста кај документи со релативно едноставен изглед и нивната примена е ограничена на документи со една колона, иста големина на букви и со релативно чисти маргини.

Garris и Grother [GG95] за проценка на закосеноста на пополнети формулари ја користат т.н. функција на закосеност:

$$S(\theta) = \sum_{i=1}^{n} e^{p_i / q_i} \tag{4.13}$$

каде p_i е сумата на црни пиксели по должината на i-тата паралелна траекторија налегната за агол θ, а q_i го претставува очекуваниот број на црни пиксели по должината на линијата кој се добива со делење на вкупниот број на црни пиксели на целата слика со висината на сликата. Со максимизација на оваа функција се добива глобалниот агол на закосеност на формуларот.

Saitoh и Pavlidis [SP92] го одредуваат аголот на закосеност по одвојувањето на поврзаните компоненти со поставување на линија по методот на најмали квадрати низ центрите на рамките околу поврзаните компоненти (blobs). Baird [Bai87] врши проекција на средишните точки на долниот раб на рамките околу поврзаните компоненти под различни агли на линија од акумулатори нормална на правецот на проекција од кои се пресметува т.н. мерка на енергијата на поравнетост (energy alignment measure) чиј максимум го дава аголот на закосеност. Пријавена е постигната точност при проценката на аголот во рамките од само 2´. Fletcher и Kasturi [FK88] по генерирањето на поврзаните компоненти и нивното филтрирање за да ги издвојат само оние кои најверојатно претставуваат букви и

применуваат Hough трансформација [Dav90, Sch89] на нивните тежишни точки. Потоа со изнаоѓањето на максимумите во Hough доменот ги пронаоѓаат локациите и закосеноста на низите од букви. Некои истражувачи [SYT94, OG92, HYR86] го користат фактот дека растојанието меѓу соседните знаци е многу помало по должината на редовите од текстот отколку помеѓу редовите за одредување на закосеноста и правилното групирање на поврзаните компоненти во редови со текст.

Hinds *et al.* [FHD90] пресметуваат Hough трансформација на намалена слика на документот, за потоа аголот на закосеност да го одредат со пронаоѓање на максимумот во Hough доменот. Yan [Yan93] го одредува аголот на закосеност испитувајќи ја интер-корелацијата меѓу вертикални линии на фиксно растојание во сликата на документот. Sauvola и Pietikäinen [SP95a] за одредување на аголот на закосеност употребуваат методи за анализа на текстури над заматена и намалена слика на документот. Последниве методи овозможуваат проценка на аголот на закосеност и на слики во нијанси на сиво, уште пред нивната бинаризација и екстракција на поврзани компоненти.

Зависно од методите за сегментација корекцијата на закосеноста може да биде различна. Кај некои методи (повеќето bottom-up методи) таа не е експлицитно видлива и е инкорпорирана во самиот процес на сегментација и групирање на компонентите. Кај top-down методите осетливи на закосеност корекцијата на закосеноста секогаш се презема пред сегментацијата. Методите кои се користат за ротација на слика се воглавно стандардни кои секој пиксел од оригиналната слика го пресликуваат на нови координати во заротираната слика. Hough трансформацијата и ротацијата на дигитализирани слики подетално ќе бидат претставени во поглавјето 5.1.

Сегментација на блокови (зонирање)

Во процесот на сегментација на блокови се врши поделба (партиционирање) на сликата на документот на неколку обично правоаголни хомогени региони. Вообичаени и најчесто користени методи за сегментација на блокови се т.н. CRLA алгоритам (Constrained Run-Length Algorithm) и RXYC алгоритамот (Recursive X-Y Cuts).

CRLA алгоритамот [WCW82, WS89] познат и како RLSA алгоритам (Run Length Smoothing Algorithm) работи со црно-бели (бинарни) слики изминувајќи ги прво ред по ред и заменувајќи ја секоја низа на последователни нули пократка од зададена вредност со низа од единици со иста должина. Потоа истата постапка се спроведува изминувајќи ја сликата колона по колона. На крајот резултатите од двете изминувања пиксел по пиксел се множат логички со што се добива т.н. "размачкана" (smeared) слика. Резултатот од RLSA алгоритамот е слика на која црните региони ги претставуваат структурните блокови на документот кои потоа се регистрираат со методата на поврзани компоненти или некоја друга метода.

RXYC алгоритамот [NSS+85] познат и како RPPC (Recursive Projecrion Profile Cuts) е метод за top-down сегментација кој сликата на документот рекурзивно ја дели на правоаголни блокови. Во секој чекор од рекурзивниот процес се

пресметуваат хоризонталната и вертикалната проекција[1] на блокот од сликата кој се дели. Поделбата вдолж двата правца се изведува со сечење (делење на блокот) на местата кои кореспондираат на длабоки долини пошироки од зададена вредност во проекциите. Слабата страна на овој метод е неговата осетливост на искосеноста на документот. RXYC алгоритамот ќе биде поопширно претставен во поглавјето 5.2.1.

Некои автори предлагаат методи за сегментација базирани на анализа на позадината на документите. Baird *et al.* [BJF90] спроведува сегментација анализирајќи ја структурата на позадината со препокривање раководено од формата (shape-directed covers). По идентификацијата на поврзаните компоненти тие се претставуваат како црни правоаголници и се спроведува корекција на закосеноста. Потоа се одредува максималното множество на бели правоаголници кои можат комплетно да ја прекријат позадината. Сегментацијата се изведува со одредување на максималното подмножесто од ова множество наречено препокривачко множество (cover set). Изборот на препокривачкото множество е воден од својствата на секој правоаголник како неговата површина и односот на неговата ширина и висина. По наоѓањето на унијата на сите правоаголници од препокривачкото множество, поврзаните компоненти од комплементот на оваа унија ја дефинираат сегментацијата. Pavlidis и Zhou [PZ91] вршат сегментација на текстови со пронаоѓање на тековите од бела површина околу текстот. Лоцирањето на белите површини се изведува со пресметување на проекциите на мали растојанија за да се намали влијанието на закосеноста и барање на долги бели региони во проекциите. Врз база на проекциите се формираат блокови кои потоа се обединуваат во поголеми и се организираат според нивната поставеност и димензии.

Saitoh и Pavlidis [SP92] предлагаат bottom-up техника за сегментација на комплексни документи без претходна елиминација на закосеноста и без претпоставка дека блоковите се правоаголни. Тие прво ја намалуваат сликата скенирана со резолуција од 400 dpi 8 пати по вертикала и 4 пати по хоризонтала, што е сеуште доволно прецизно за да не дојде до допирање на различните блокови. Потоа на ваквата слика ги лоцираат поврзаните компоненти, кои потоа ги организираат во поголеми блокови зависно од нивните димензии и содржина. Техниката е испробана на 52 јапонски и 21 англиски документи без наведување на квантитативни мерки за перформансите.

Jain и Bhattacharjee [JB92] за сегментација на документи користат Gabor филтри кои обично се употребуваат за сегментација на текстури. Применето е повеќеканално филтрирање за сегментација на текстури на тој начин што оригиналната слика на документот се пропушта низ n парно симетрични дводимензионални Gabor филтри со што се добиваат n филтрирани слики. За секоја од овие слики се пресметува "слика на својства" која ја изразува локалната енергија над мал прозорец околу секој пиксел од филтрираната слика. Потоа регионите од оваа слика се делат во три множества кои ги претставуваат

[1] Во обработката на слики под поимот проекција се подразбира пресликување од дводимензионална слика во еднодимензионална бранова форма чии вредности претставуваат сума на вредностите на пикселите вдолж одреден правец. Хоризонтална односно вертикална проекција претставува сума на вредностите на сите пиксели вдолж линија линија паралелна со y односно x оската.

регионите со текст, празните региони и регионите кој претставуваат граница меѓу нив од каде се спроведува сегментацијата.

Tang *et al.* [MXM+95] вршат сегментација на документи со анализа на сликата на документот набљудувајќи ја како фрактал. Тие дефинираат фрактален потпис (fractal signature) со анализа на кој вршат сегментација на документот користејќи го фактот дека фракталната димензија (а со тоа и фракталниот потпис) е поголема за текстуалните региони отколку за позадината и графиката.

Класификација на блокови

Целта на класификацијата на блоковите е да се процени содржината на секој од блоковите како би можел да се упати на соодветна обработка или да се игнорира. Во процесот на класификација блоковите добиени во процесот на сегментација се класификуваат во една од релативно малиот број предодредени категории како: текст, слика, позадина, … Kaj голем број bottom-up методи сегментацијата и класификацијата на блоковите се обединети во единствена неделива постапка или се одвиваат паралелно.

За класификација на блоковите најчесто се користат статистички методи кои врз база на некои својства на блоковите ги распоредуваат во некоја од предодредените категории. Својствата кои најчесто се користат за класификација на блоковите се: нивните димензии, ориентација, односот на бели и црни пиксели, просечните и максималните должини на низите од последователни црни или бели пиксели, или својствата во трансформациониот домен на трансформацијата користена при сегментацијата. Kaj поголемиот дел на bottom-up методите класификацијата на блоковите е интегрирана со процесот на сегментација. Bottom-up методите за сегментација најчесто уште по првата итерација на генерирање на блоковите од најниско ниво (поврзани компоненти или прозорци со фиксни димензии) вршат класификација на ваквите блокови, па дури потоа ги обединуваат во поголеми, користејќи го при тоа и податокот за припадноста на секој од блоковите кон некоја од предодредените категории.

Wang и Srihari [WS89] вршат сегментација на скенирани страни од дневен печат со примена на CRLA и RXYC алгоритамот. Вака сегментираните блокови потоа се класификуваат во една од класите: полутонови (фотографии), текст со големи букви (наслови), текст со средни букви (воведници), текст со мали букви и линиска графика. За класификација на блоковите се употребува анализа на текстурата која се спроведува во два чекора. Во првиот чекор се пресметуваат посреднички матрици (intermediate matrixes) врз база на некои особини на блоковите. Во вториот чекор од овие матрици се издвојуваат некои особини кои се користат како влез во статистички класификатор. Се користат две посреднички матрици: BW матрица (black-white pair run length matrix) која ја мери должината на секој пар соседни низи од последователни црни и последователни бели пиксели и BWB матрица (black-white-black pair run length matrix) која ја мери должината на секоја тројка соседни низи од последователни црни, па бели па повторно црни пиксели, каде должините на низите од црни пиксели е приближно иста. Од BW матрицата се изведуваат две особини кои ги истакнуваат кусите и долгите низи соодветно. Третата особина се изведува од BWB матрицата и ги истакнува многу долгите низи. Во просторот на особини формиран само од

првите две особини добро може да се класификуваат полутоновите и сите големини на букви, но не и линиската графика за чија правилна класификација е неопходно и третото својство изведено од BWB матрицата.

Doerman *et al.* [Doe94, ECD94, EDC94, EDC95] изложува метода за сегментација и класификација со помош на wavelet пакети и интегрирано одлучување. Пирамидалната wavelet трансформација во форма на wavelet пакети е употребена за добивање на вектор со особини за мал прозорец од сликата на различни нивоа на резолуција. Векторите со особини од различните нивоа на резолуција се влез во повеќенивовска невронска мрежа која го класификува прозорецот во една од класите текст, слика или ниту текст – ниту слика. Излезот од невронската мрежа се користи во процесот на "непрецизна класификација" (soft classification) за одредување на конечната припадност на секој прозорец со помош на тежинско гласање од соседните прозорци на исто ниво на резолуција, како и од прозорците на други нивоа на резолуција. На крајот блоковите од иста класа се обединуваат во поголеми.

Sauvola и Pietikäinen [SP95b] предлагаат метода за сегментација и класификација со брзо екстрахирање на особини и класификација со помош на правила. Kaj овој пристап сликата прво се дели на прозорци со еднакви димензии (од 10×10 до 20×20 пиксели) и за секој од нив се пресметуваат четири основни особини: односот на бели и црни пиксели во него, просечната должина на низите од последователни црни пиксели, вертикалната меѓу-корелација меѓу соседните пиксели и меѓу првиот и секој петти пиксел. Секој од прозорците потоа со помош на правила за одлучување се класификува во една од класите: текст, графика или позадина. Вака добиената шема од прозорци итеративно се филтрира со помош на специјални 3×3 и 4×4 маски по што, повторно со употреба на соодветни 3×3 маски се детектираат аглите на униформните блокови кои ја претставуваат сегментацијата.

Повеќето методи во процесот на сегментација грубо ги класификуваат блоковите во мал број категории – најчесто текст, слика и позадина, за по конечната сегментација доколку е потребно да се спроведе детална класификација. Така на пример одредувањето дали текстуалниот блок содржи печатен или ракописен текст е неопходно заради негово проследување на соодветен алгоритам за сегментација на знаците и препознавање. Метод за одлучување дали даден текст е печатен или ракописен, употребен за обработка на поштенски адреси, предложен од Srihari *et al.* [SL+92] се базира на пресметување на хистограм на височините на поврзаните компоненти. За разлика од печатените текстови кои обично имаат само неколку изразени пикови во хистограмот, широката дистрибуција на височините на поврзаните компоненти упатуваат на ракопис. Srihari *et al.* [SSR94] во апликација за препознавање на имиња и адреси од порезни формулари, употребува Fisher-ов линеарен дискриминатор за класификација на текстовите како печатени или ракописни. Дискриминаторот користи шест симболички особини изведени од поврзаните компоненти: стандардни девијации на ширината и висината на поврзаните компоненти, просечна густина на компонентите, однос на висината и ширината и одредени различни висини и ширини на поврзаните компоненти.

Блоковите со слика можат да припаѓаат на една од следниве категории: слика со полутонови, црно-бел цртеж (дијаграм) и табела. Сликите во нијанси на сиво, односно со полутонови лесно се одвојува од блоковите со дијаграми и

табели со анализа на хистограмот на дистрибуција на нијанси на сиво, или преку просечната должина на низи од последователни црни пиксели. Иако и дијаграмите и табелите се состојат претежно од прави линии и текст, фактот дека кај табелите се јавуваат само хоризонтални и вертикални линии може да се искористи при одредувањето дали се работи за дијаграм или табела.

Орѓанизација на блокови

Задачата на организацијата на блоковите е да се воспостави соодветна релација меѓу физичката и логичката структура на документот. Ова е неопходно за правилна интерпретација на одделните блокови (наслов, главен текст, или адреса, тело, потпис) и одредување на редослед по кој резултатите од препознавањето ќе бидат презентирани на корисникот.

Организацијата на блоковите обично се спроведува користејќи правила изведени од познавањата за форматот на документот кој се обработува или некои глобални правила кои се применуваат при формирањето на документи (обично насловот е поставен над телото на текстот и е со поголеми букви), како и положбата, димензиите и некои дополнителни атрибути на блоковите добиени во процесот на сегментација и класификација. Базата на знаење потребна за успешна организација на блоковите е доста специфична и може да биде изведена како експлицитни if-then правила за одлучување, јазик за опис на форми (form-definition language), геометриско стебло од блокови, експертен систем, формална граматика или да биде имплицитно вклучена во процесот на декомпозиција на документите.

Од аспект на оптичкото препознавање на знаци организацијата на блокови опфаќа одредување на редоследот при читањето на текстуалните блокови. Кај документите со сложен изглед (како страниците на часописите, рекламите, проспектите) редоследот на читањето не е секогаш единствен и неговото одредување не е тривијална работа. Некои системи конечната организација на блоковите ја одредуваат дури по препознавањето на текстот и извлекување на некои семантички информации од него кои ги користат при конечното одредување на редоследот.

Во апликација за препознавање на деловни писма, [Den90, DBH+92] Dengel организацијата на блоковите ја спроведува со употреба на база на знаење. Специјален модел на изглед на деловно писмо наречен геометриско стебло се користи за генерирање на хипотези за логичкото значење на геометриските блокови од документот. За тестирање на хипотезите се користи дополнителна статистичка база на податоци во која се сместени сите можни логички објекти и нивните геометриски особини кои се среќаваат кај деловните писма. Користена е стратегија на генерирање на хипотези и нивно тестирање со која се избегнува препознавање на текстот пред организацијата на блоковите. Гранењето низ геометриското стебло кое води кон соодветната организација на блоковите е раководено од мерки на сличност кои се пресметуваат со користење на систем на правила и податоците од статистичката база.

Saitoh *et al.* [SYT94] градат граф од блоковите во кој воспоставуваат родител-дете релација. Оваа релација се воспоставува врз база на просторниот распоред на блоковите според која родителот има “влијание” над детето. Влијанието на даден јазол се наследува од неговите потомци и се пропагира

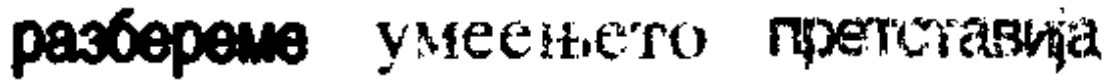
надолу, но е ограничено од вертикалните линии и другите блокови. Од овој граф се гради стебло испитувајќи ја "областа на влијание" (influence range) на блоковите и пронаоѓањето на соодветен родител за секој блок. На крајот, редоследот на читање се добива со preorder изминување на стеблото.

Nagy *el al.* [NSV92] употребува блокови граматики (граматики за опис на блокови) за организација на блоковите при сегментација на страници од научни трудови. Секоја блокова граматика вклучува знаења за правилата на организација на страницата кои се користат за поделба на блокот на подблокови. Множество од блокови граматики формира документ граматика (граматика за опис на документи) со која страниците се делат на вгнездени правоаголни блокови и се претставуваат со податочна структура наречена X-Y стебло.

За проценка на точноста на автоматската сегментација и одредување на редоследот на читање на документите присутни се воглавно два пристапа. Saitoh *et al.* [SYT94] проценката на точноста на сегментацијата ја изведуваат врз база на бројот и видот на интервенции од страна на операторот потребни за да се поправи сегментацијата. Kanai *et al.* [KRN+95] предлагаат метода со која точноста на автоматското зонирање се проценува според бројот на потребни интервенции над препознатиот текст добиен како излез од идеален OCR, за тој да се доведе во правилен редослед. Тие воведуваат т.н. метрика на зонирање која се пресметува врз база на минималниот број на операции на бришење, вметнување и преместување за секоја од кои е дефинирана различна цена на чинење, потребни излезот од OCR програмата да се трансформира во коректниот текст. Garris [Gar95] предлага метода за проценка на зонирањето врз база на споредување на хипотезите за положбата на зоните добиени при автоматското зонирање со вистинските референтни положби на зоните (внесени од операторот). Дефинирана е мерка за изразување на растојание помеѓу две зони како и мерка за сличност помеѓу зоните, преку кои се изразува грешката при зонирањето. Изложени се два различни пристапа: едниот базиран на преклопување на зоните кој е посоодветен за поврзаните зони, додека другиот е базиран на поравнетоста на зоните и е посоодветен за вгнездени зони. Taghava *et al.* [TCB95] за проценка на целосното зонирање, препознавање и опис на документот користи споредување на излезната слика генерирана од описот на документот со оригиналната.

4.4.7 Сегментирање на знаци

Сите пристапи за оптичко препознавање на знаци вршат сегментација на сликата на текстот најмалку до ниво на изолирани зборови. Кај печатените документи зборовите се составени од поедини изолирани знаци кои во идеален случај не се допираат. Растојанието меѓу буквите во состав на еден збор е доста помало од растојанието меѓу соседните зборови и редови со текст, па соодветно евентуалните деформации на документот тука стануваат најизразени. Изобличувањата кои настануваат при печатењето и вообичаената манипулација со печатени материјали се најчесто од видот заматување и бледење кои предизвикуваат допирање на соседните букви односно нивно соединување во единствена компонента или губење на некои нивни делови и распаѓање на знакот на повеќе одвоени компоненти (Слика 4.6). Соодветно, изолирањето на поединитe букви од сликата на зборот претставува нетривијален проблем кој посебно се разгледува.



(а) (б) (в)

Слика 4.6 Типични изобличувања на печатен текст: (а) соединување (допирање) на соседните знаци (touching characters); (б) распаѓање на знаците (broken characters); (в) истовремено допирање и распаѓање на знаците

Сегментацијата на знаци претставува процес на делење на сликата со секвенца од знаци на подслики на индивидуални знаци. Процесот на сегментација на знаците е еден од процесите на одлучување во системот за OCR. Одлуката дали одреден дел од сликата претставува слика на изолиран знак, може да биде точна или погрешна, а при препознавањето на неквалитетно печатени документи таа е погрешна толку често, што претставува главна причина за слабите резултати во препознавањето.

Кај класичниот пристап за оптичко препознавање на знаци, сликата на низата од знаци (зборот) се дели на подслики од кои секоја претставува слика на еден изолиран знак, која потоа се упатува на препознавање. Поделбата на изолирани знаци го наметнува прашањето: "Што го сочинува знакот?". Од аспект на оптичкото препознавање на знаци знакот претставува примерок **сличен** на некој од знаците за чие препознавање е дизајниран системот. За да се оцени сличноста пак, сликата на примерокот мора да се **изолира** од остатокот на документот, со што се доаѓа до парадоксална ситуација во која секој од чекорите зависи од другиот. Исто така одлуката за сегментација на знак не е локална одлука независна од претходните и наредните одлуки. Постигањето на добро совпаѓање на изолираниот знак со некој од примероците на симболи од базата е потребен но не и доволен услов за сигурно препознавање. Кај оштетените документи лесно може да се случи поради изобличеност која води кон погрешна сегментација одредени знаци повеќе да личат на некои други и како такви и да бидат препознати, но ваквите резултати од препознавањето често не претставуваат контекстно валиден резултат. На пример на Слика 4.7 'десетти' зависно од сегментацијата би можело да биде препознато и како 'деоетги', а 'уапси' како 'уагки', но ваквото препознавање не води кон контекстно валиден резултат. Кај оштетените и изобличени документи сегментацијата е поголем извор на грешки отколку самата изобличеност на знаковите [RKN94, RKN95].

(а) (б)

Слика 4.7 Оштетеноста на сликата условува неправилна сегментација која води кон погрешно препознавање

Зависно од интеракцијата меѓу сегментацијата и препознавањето, методите за сегментација на знаци може да се поделат на три "чисти" пристапа и голем број на хибридни стратегии кои претставуваат комбинација на овие три. Трите елементарни стратегии се:

- дисекција;

- сегментација на база на препознавање; и
- холистички[1] пристап.

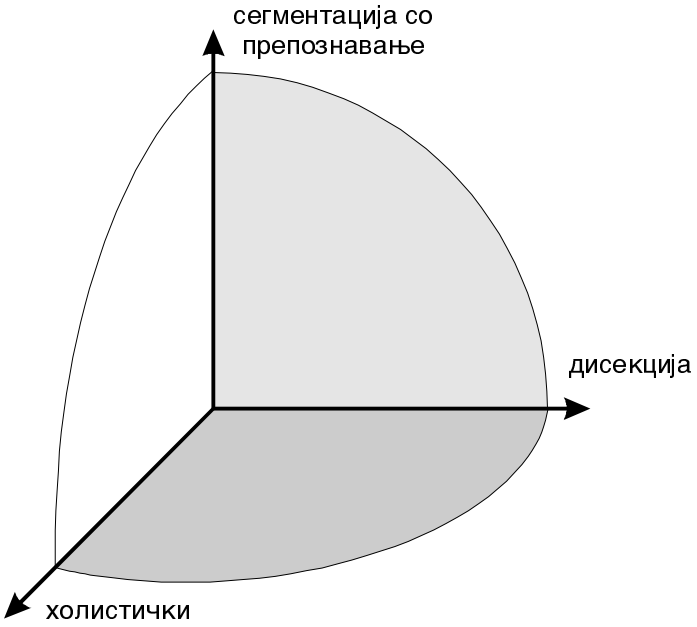
Процесот на делење на сликата на подслики од изолирани знаци врз база на некои нивни карактеристики е наречен дисекција. Карактеристично за ваквите методи е што сегментацијата на знаци секогаш претходи на нивното препознавање и меѓу нив постои јасно воочлива граница. Кај оваа класа методи критериумот за делење на знаци е совпаѓањето на некои генерални особини на сегментите со очекуваните особини на валидно изолирани знаци. Особини кои обично се употребуваат за сегментација на знаци се: висината, ширината, одвоеноста од соседните компоненти, поставеноста во однос на основната линија и др.

При сегментацијата на база на препознавање, системот ја пребарува сликата барајќи компоненти кои можат да бидат препознати како валидни знаци. Критериумот за делење на знаци кај овие методи е степенот на препознавање, вклучувајќи и евентуална синтактичка и семантичка проверка на резултатот. Кај ваквите методи сегментацијата и препознавањето се одвиваат паралелно. Методите за сегментацијата на база на препознавање од своја страна може да се поделат во две подкатегории. Методите за сегментирање со лизгачки прозорец (sliding window) слепо ја делат сликата на зборот на многу места не земајќи ги при тоа предвид карактеристиките на сликата и потоа се обидуваат да изнајдат оптимална сегментација евалуирајќи ја успешноста во класификацијата на генерираните подслики. Методите за сегментирање врз база на особините, ја детектираат позицијата на одредени особини на сликата и се обидуваат да ја поделат на препознатливи подслики користејќи ги при тоа нејзините особини.

Холистичкиот пристап всушност и не претставува сегментирање на знаци туку метод тоа да се избегне, препознавајќи цели зборови без потреба од претходно изолирање на буквите.

Најголемиот дел од методите кои се употребуваат за сегментација на знаци се хибридни методи кои претставуваат некаква тежинска комбинација на претходните три методи. Ако трите фундаментални стратегии се претстават како три ортогонални оски, тогаш хибридните методи може да се претстават како тежинска комбинација од нив која претставува точка од просторот зафатен со нив (Слика 4.8). Овој простор е континуален и постои само концептуално т.е. нема смисла да се одредуваат точните тежински учества на секоја од фундаменталните стратегии за дадена хибридна метода.

[1] Од holism – теорија според која природата претставува единство на деловите кои ја сочинуваат целината и е многу повеќе од прост збир на деловите.

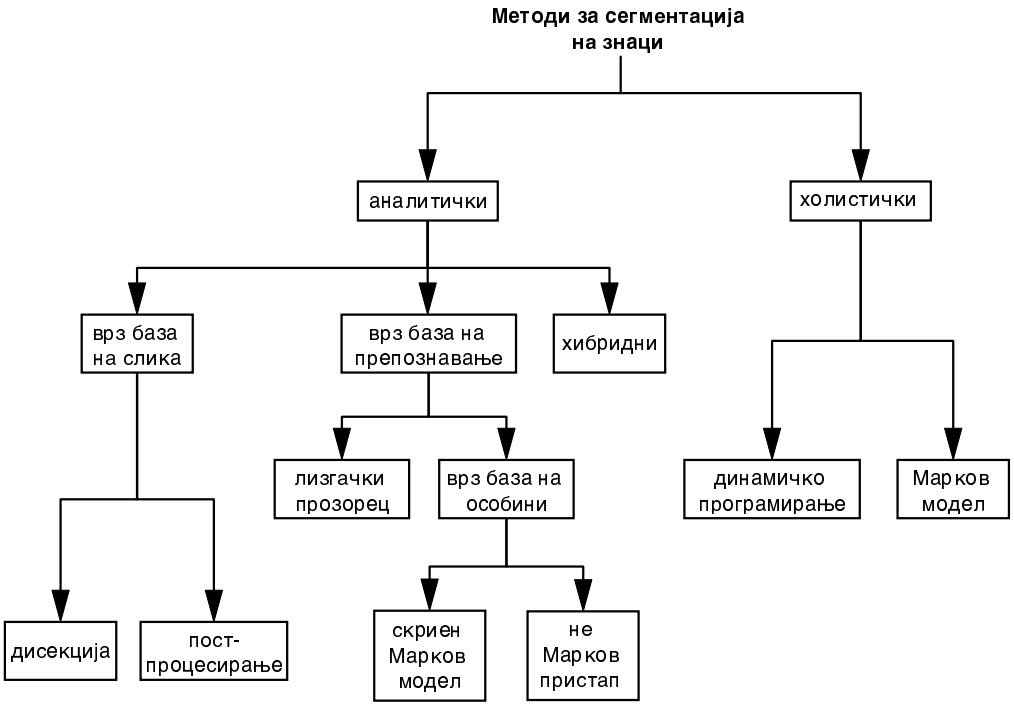


Слика 4.8 Просторот на стратегиите за сегментација на знаци

Според Casey и Lecolinet [CL96] методите за сегментација на знаци може глобално хиерархиски да се поделат како на Слика 4.9.

Дисекција

Дисекцијата претставува процес на декомпозиција на сликата во секвенца на подслики врз основа на некои генерални одлики на сликата. За разлика од методите со лизгачки прозорец кои сликата ја делат независно од нејзината содржина, кај дисекцијата содржината на сликата, односно особините извлечени од неа, имаат клучна улога при поделбата на сликата. Дисекцијата претставува интелигентен процес во кој се спроведува анализа на сликата, но не и препознавање на симболите. Кај системите кај кои сегментацијата и препознавањето не се во интеракција, дисекцијата го претставува целосниот процес на сегментација на знаци. Меѓутоа, сегментацијата претставува комплексен процес, па често пати под дисекција се подразбира процесот на физичкото делење на сликата на подслики, како подпроцес на целосната сегментација на знаци која може да вклучува и контекстно знаење или проценка на формата на знаците.



Слика 4.9 Хиерархиска поделба на методите за сегментација на знаци

Пионерските обиди за оптичко препознавање на знаци во 50-ите и 60-ите години биле концентрирани кон препознавањето на изолирани знаци. Употребата на специјални формулари, фонтови со зголемено растојание меѓу буквите и специјално дизајнирани знаци (Слика 2.2) кај кои сегментацијата не претставувала особен проблем, а која била услов за деловна примена на OCR технологијата, го објаснува фактот што сегментацијата е ретко споменувана во литературата пред 70-ите години.

Кај печатените текстови вертикалната празнина редовно се користи како сепаратор меѓу последователните знаци. Идејата за детекција на вертикалните празнини помеѓу последователните знаци се наметнува како природен концепт при дисекцијата на печатени текстови. Од аспект на широчината, знаците кај печатените текстови можат да бидат со фиксна или пропорционална широчина (Слика 4.10). Испис со знаци со фиксна широчина даваат машините за пишување, линиските и постарите матрични печатачи. Кај ваквиот испис секој знак зафаќа место со еднаква широчина. Пропорционалниот испис е својствен за печатарските машини и кај него широчината на знаците не е фиксна т.е. секој знак зафаќа различно место зависно од неговата форма. Така на пример буквите ‘ш‘ или ‘М’ се неколкупати пошироки од знакот ‘!’ или буквата ‘ј’.

(а) (б)

Слика 4.10 Примери на текст со (а) фиксна и (б) пропорционална широчина

Во зависност од употребеното писмо, од аспект на дисекцијата печатените текстови може да се поделат на:

- Нормален текст;
- Текст од знаци кои се преклопуваат но не се допираат (кернирани);
- Курзивен текст – *италик*;
- Текст од знаци кои се допираат.

Во секојдневната употреба најчесто се користат фонтови кај кои знаците не се допираат и не се преклопуваат, како на Слика 4.11 а. Кај ваквите текстови хоризонталниот простор што го зафаќа секој од знаците не се преклопува со соседните. За разлика од ова, кај некои фонтови знаците иако не се допираат со некој свој дел навлегуваат и во хоризонталниот простор на соседните знаци, односно нивните хоризонтални простори се преклопуваат како на Слика 4.11 б. При употребата на курзивни писма кај кои знаците се искосени под некој агол доаѓа до природно преклопување на нивните хоризонтални простори. Кај некои (декоративни) фонтови некои од знаците се допираат слично како кај ракописот па за нивна сегментација се користат методи карактеристични за сегментација и препознавање на ракописни текстови.

(а) (б) (в) (г)

Слика 4.11 Пример за: (а) нормален текст, (б) текст од знаци кои се преклопуваат, (в) курзивен текст, (г) текст од знаци кои се допираат

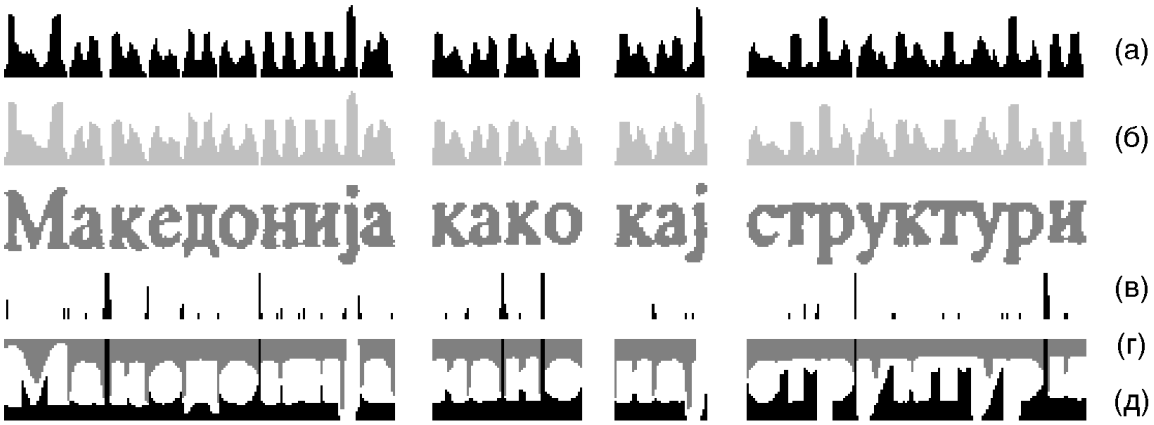
При сегментацијата на текстови од знаци со фиксна широчина, бројот на знаци по единица хоризонтално растојание кој е приближно константен претставува база за проценка на сегментационите точки. Сегментационите точки за даден ред со текст би требало да се подеднакво оддалечени една од друга. Ова обезбедува база за глобална проценка на сегментацијата бидејќи сегментационите точки не се независни. Примената на ова правило обезбедува правилно сегментирање и во случаите на споени или распаднати знаци. Доколку најголемиот дел од дисекции може да се направи со пронаоѓање на колони од бели пиксели, секвенцата на граници на сегментационите региони може да се процени. Сегментационите точки кои не лежат блиску до овие региони можат да бидат отфрлени како резултати на распаднати знаци, а сегментационите точки кои недостасуваат поради споените знаци може да се проценат и да се спроведе локална анализа за изнаоѓање на најпогодно место за дисекција.

Hoffman и McCullough [HM71] го формализираат процесот на сегментација на знаци и во нивната формулација тој е составен од три чекори:

1. Детекција на почеток на знак;
2. Одлука да се почне со барање на крај на знак (sectioning);
3. Детекција на крај на знак.

Вториот чекор е критичен во процесот на сегментација и се базира на тежинска анализа на хоризонталните низи од црни пиксели кои завршиле наспроти оние кои уште не се завршени, испитувајќи го редот со текст колона по колона. Проценката на просечната широчина на знаците исто така е параметар во процесот. По пронаоѓањето на регион во кој е дозволена сегментација, таа се врши или со примена на правила за детекција на зголемена густина на пикселите

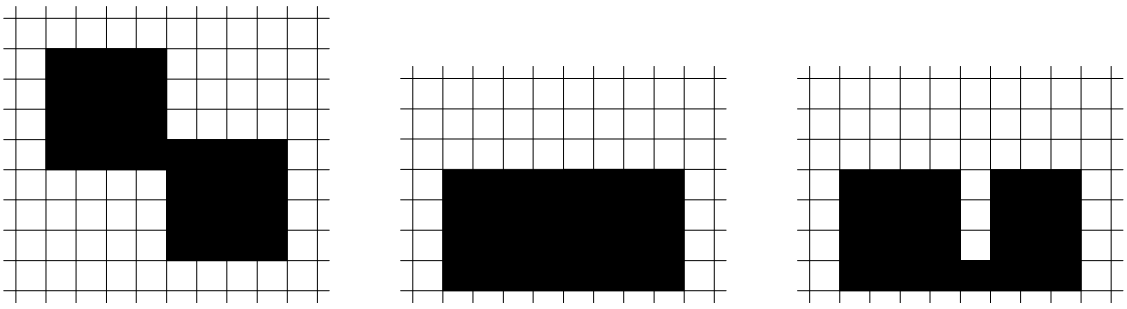
(почеток на нов знак) или испитување на некои особини специјално дизајнирани за детекција на крај на знак. Авторите пријавуваат дека на множество од 80.000 знаци од кои 22% допрени, сегментацијата била спроведена со точност од 1 пиксел во околу 97% од случаите.



Слика 4.12 Проекции кои се користат при дисекција: (а) break cost function; (б) вертикална проекција; (в) однос на втората диференција и вредноста на вертикална проекција; top profile (горен профил); (д) bottom profile (долен профил)

Анализата на *вертикалната проекција* (наречена и вертикален хистограм) е една од најчесто користените методи за сегментација на знаци. Вертикалната проекција на ред со текст (Слика 4.12 б) се состои од едноставно броење на црните пиксели во секоја колона. Таа може да послужи за пронаоѓање на празнините помеѓу соседните знаци како и позициите на вертикални линии кај печатените знаци. Во случаите кога печатените знаци се допираат или хоризонтално се преклопуваат, вертикалната проекција обично има минимум во колоната каде треба да се изврши сегментацијата. Некои истражувачи [BKP86] го користат односот на втората диференција и вредноста на самата проекција во таа колона (Слика 4.12 в) како критериум за одбирање на колони за сепарација. Оваа функција има изразени пикови во минимумите на проекцијата и со нејзина примена се избегнуваат проблемите на сегментација во областите на тенки хоризонтални линии. Lu [Lu95] ја користи т.н. *peak-to-valley* функција за сегментација на допрени знаци. При конструкцијата на оваа функција прво се бара локален минимум во низата на вертикални проекции, а потоа локалните максимуми лево и десно од него. Односот на сумата на диференциите меѓу минимумот и соседните максимуми и самата вредност на минимумот (плус 1, за да се избегне делењето со нула) ја претставуваат вредноста на peak-to-valley функција. Оваа функција се карактеризира со изразени пикови во колоните кои во низата на вертикални проекции имаат минимална вредност опкружена со високи вредности од двете нејзини страни. Tsujimoto и Asada [TA92] ја употребуваат т.н. *break cost function* за оценка на допреноста на соседните знаци. Оваа функција е дефинирана за секоја позиција меѓу две соседни колони од сликата и претставува број на црните пиксели добиени со AND операција помеѓу соседните колони (Слика 4.13). Break cost функцијата фактички претставува вертикална проекција во која при броењето на пикселите во секоја колона, се бројат само црните пиксели кои непосредно лево од себе исто така имаат црн пиксел. Вака конструираната функција дава подлабоки долини меѓу допрените знаци од обичната вертикална проекција (Слика 4.12 а). Kimura и Shridar [KS92] за

сегментација на знаци ја употребуваат вертикалната проекција, но изборот на најпогодна колона за разделување на знаците кои се допираат се прави според критеруим базиран на односот на расејувањата меѓу класите и во рамките на класата, дефинирани преку некои основни статистики на проекциите. Некои истражувачи ги користат и т.н. *горен* и *долен профил* (top/upper profile, bottom/lower profile) при сегментацијата на знаци. Функцијата на горен профил е дефинирана како максималната *y*-вредност на црните пиксели од секоја колона на сликата (Слика 4.12 г). Соодветно долниот профил е дефиниран како минималната *y*-вредност на црните пиксели од секоја колона на сликата (Слика 4.12 д). Dabrowska и Depowska [DD95] претставуваат метод за сегментација на допрени и знаци кои се преклопуваат базиран на анализа на горниот и долниот профил.



(а) (б) (в)

Слика 4.13 Пример за break cost функција: (а) слика на две допрени области; (б) вертикална проекција на (а); (в) break cost функција на (а)

Изолирање и обединување на поврзани компоненти

Kaj bottom-up пристапите за сегментација на сликата на документот, прво се врши сегментација до ниво на поврзани компоненти а потоа нивно обединување во хиерархиски целини. Меѓутоа, меѓу поврзаните компоненти и знаците не важи строга еден спрема еден кореспонденција. Постојат знаци (‘ј’, ‘ѓ’, ‘ќ’, ‘;’, ‘?’, …) кои природно се составени од повеќе поврзани компоненти, додека допрените знаци ќе бидат регистрирани како единствена поврзана компонента. До распаѓање на знаците на повеќе поврзани компоненти може да дојде и поради изобличувања на документот. Знаците составени од повеќе компоненти мора да бидат обединети во единствени ентитети пред препознавањето. Обединувањето на поврзаните компоненти е раководено од критериуми изведени врз база на меѓусебната положбата на поврзаните компоненти (најчесто се вертикално поравнети, односно значително се преколопуваат по хоризонтала), нивните димензии, односот на висината и ширината и пополнетоста на рамката околу поврзаната компонента со црни точки. Користејќи слични критериуми се врши пронаоѓање на поврзаните компоненти составени од повеќе допрени знаци за чие раздвојување се користат слични методи за сегментација на знаци како и кај top-down методите (најчесто дисекција врз база на вертикалната проекција или некоја друга функција изведена од неа). Пристапот за сегментација на знаци по пат на поврзани компоненти претставен од Spitz [Spi94] ги користи информациите за вертикалното простирање на поврзаната компонента во однос на редот со текст при раздвојувањето на допрените и обединувањето на знаците кои се состојат од повеќе компоненти.

Целта на сегментацијата обично е да се раздели сликата на текстот на низа од подслики од кои секоја содржи еден знак. Алтернативниот пристап базиран на дисекција е разделување на сликата на подслики кои не мора да бидат индивидуални знаци. Ваквата дисекција се спроведува врз база на некои карактеристики на сликата кои се појавуваат меѓу соседните знаци, но можеби и во рамките на индивидуален знак. Ваквите форми наречени графеми или псевдо-знаци потоа полесно се класификуваат во одреден број класи. Процесот на препознавање го довршува контекстна функција на пресликување од множеството на класи од графеми во множеството на симболи. При ова овозможено е еден-во-многу и многу-во-еден пресликување, односно функцијата на пресликување може да изврши разделување или обединување на графеми. Чекорот на дисекција кај ваквите пристапи е наречен пресегментација (presegmentation) или кога целта е да не се остави целосен знак – over-segmentation.

Сегментацијата на знаци по пат на дисекција може по препознавањето да биде проценета врз база на резултатите од препознавањето и лингвистичкиот контекст и по потреба да се направи алтернативна сегментација. Во еден поинаков пристап [LM89] базиран на концептот на "пред-препознавање" имплементирана е техника која ја следи анализата на поврзаните компоненти со едноставна логика за препознавање чија цел не е да се препознае знакот, туку само да се одреди кои од поврзаните компоненти се единствени знаци, допрени знаци или распаднати знаци.

Сегментација на знаци базирана на препознавање

Значајна гранка од аналитичките методи за сегменација се методите базирани на препознавање. И овие методи ја сегментираат сликата на зборот на подслики од индивидуални знаци, но на доста поинаков начин. Во принцип, не се користи никаков дисекционен алгоритам врз база на својствата на сликата. Кај овие методи сликата систематски се дели на многу делови кои се преклопуваат, независно од нејзината содржина. Ваквите парчиња се проследуваат на препознавање во обид да се изнајде кохерентен резултат во препознавањето и сегментацијата. Системите кои го следат овој принцип – изведуваат сегментација на база на препознавање (recognition-based segmentation), сегментацијата е низ-продукт на препознавањето на знаци, кое од своја страна може да биде водено и од анализа на контекстот. Иако некои автори ваквите методи ги нарекуваат segmentation-free тие тоа не се, бидејќи со избегнување на сегментацијата всушност се избегнува само имплементација на посебен модул за дисекција, додека сегментацијата и понатаму имплицитно е содржана во самото препознавање. За разлика од нив холистичките пристапи навистина ја избегнуваат сегментацијата препознавајќи цели зборови.

Основниот принцип на овие методи е употребата на лизгачки прозорец со променлива широчина за добивање на пробна сегментација која се потврдува или отфрла зависно од резултатите на препознавањето. Со промената на широчината и положбата на прозорецот се добиваат повеќе различни секвенци на можни сегментации од кои секоја се проценува како целина врз база на препознавањето. Препознавањето може да се одвива или според сериската или според паралелната оптимизациона шема. Во првиот случај препознавањето се одвива од лево на десно низ зборот барајќи "задоволителен" резултат од препознавањето. Паралелната метода имплементира поглобален пристап

генерирајќи решетка од сите (или многу) можни слика-во-симбол комбинации. Финалната одлука се донесува преку изнаоѓањето на оптимална патека низ решетката. Лизгачкиот прозорец може да се движи директно низ сликата или пак да биде имплементиран во вид на тежини или групи од позициони одлики екстрахирани од сликата.

Во основа сите системи за сегментација на знаци базирани на препознавање ги имаат следниве два чекора:

1. Генерирање на сегментациони хипотези (windowing step),
2. Избор на најдобрата од хипотезите (verification step),

но начинот на нивното изведување се разликува од систем до систем.

Веројатно првата теоретска и експериментална примена на овој концепт е пријавена од Kovalevsky [Kov68] за препознавање на кирилични знаци печатени со слаб кавалитет. Применет е лизгачки прозорец кој се движи по должината на редот со текст, при што сликата која се наоѓа под прозорецот се проследува на препознавање. Моделот на Kovalevsky претпоставува дека веројатноста за набљудување верзија на прототипот на знак е сферно симетрична функција од разликата меѓу двете слики. Тогаш бараната оптимална функција на сегментација претставува сума на квадратите на растојанијата меѓу сегментираните слики и соодветните прототипови. Множеството на сегментирани слики кое ја минимизира оваа сума ја претставува оптималната сегментација. Kovalevsky забележал дека парчињата од два соседни знака кои истовремено доаѓаат под прозорецот можат да личат на трет симбол и да предизвикаат погрешно препознавање, па предлага проценката на сегментацијата да се врши по препознавање на цели зборови или редови наместо изолирани знаци. Сличен пристап реализираат и Garris и Wilson [GW92] кои со помош на невронска мрежа вршат паралелна сегментација и препознавање на знаци. Тие експериментираат со различни невронски мрежи: стандарден повеќенивовски перцептрон (Multi-Layered Perceptron – MLP) и самоорганизирачко асоцијативно пресликување, како и со единствена и одвоени невронски мрежи за сегментирање и препознавање.

Примерок	Прозорец	Прототип 1	Остаток	Прототип 2
rn	rn	m	ı	
	rı	n	n	
	rı	o	m	
	r	r	m	m

Слика 4.14 Рекурзивна сегментација

Casey и Nagy [CN82] предлагаат рекурзивен алгоритам за сегментирање на печатени знаци. Нивниот алгоритам, базиран на совпаѓање со прототипови, систематски ги тестира сите комбинации на дозволени сегментации сé додека не го испита целото множество од можни сегментации или додека не пронајде прифатлива сегментација. За прифатлива сегментација се смета онаа во која секој сегментиран примерок се совпаѓа со некој прототип од базата со одредена

толеранција. Кога сликата под прозорецот е препозната, процедурата рекурзивно се применува над остатокот, сé додека не се пронајде комплетна секвенца на успешно препознати прозорци (Слика 4.14).

Сегментација и препознавање со користење на Маркови вериги

Некои истражувачи [Elm96] ги применуваат Марковите вериги при сегментацијата и препознавањето на знаците. Скриениот Марков модел (Hidden Markov Model – HMM) ги моделира варијациите кај печатениот или ракописниот текст како скриена веројатносна структура која не е директно видлива. Оваа структура се состои од множество на состојби и веројатности на премин (transition probabilities) помеѓу состојбите. Својствата на сликата се набљудува како секвенца случајни величини чија дистрибуција зависи од состојбата.

Примената на Марковите вериги во сегментацијата и препознавањето на ракописни и печатени текстови, од аспект на особините кои ги обработува може да се подели во три групи:

1. Марковата верига ги претставува варијациите меѓу соседните знакови (типично меѓу буквите) во јазикот. Ваквите модели се базирани на фреквенциите на биграмите или триграмите во јазикот. Особините се екстрахираат од индивидуалните знаци кои претходно мора да бидат сегментирани.
2. Марковата верига се употребува за моделирање на транзициите меѓу соседните состојби во рамките на знакот. Зборот е претставен како низа од модели на знакови од кои секој е претставен со повеќе состојби. Кај ваквите пристапи знаците (имплицитно) се сегментираат при споредувањето на конкретната низа на особини екстрахирани директно од сликата на зборот со моделот на знакот, при што се донесува одлуката каде во низата од особини одреден модел на знак завршува и почнува следниот.
3. Марковата верига ги претставува варијациите меѓу состојбите во рамките на зборот кој е член на лексиконот на дозволени зборови. Ваквиот пристап е холистички и не бара ниту експлицитно ниту имплицитно сегментирање на знаците и препознава цели зборови.

Практично, само Марковите вериги од втората група вршат сегментација на знаците како низпродукт на препознавањето. Низата од особини добиени со екстракција од сликата на зборот може да се подели во изолирани знаци на повеќе начини. Веројатноста дека одредена сегментација одговара на даден збор претставува производ на веројатностите дека секој од сегментите одговара на одредена буква во рамките на зборот. Соодветно може да се пресмета веројатноста за секој од кандидатите зборови кои се добиваат со различните сегментации. При изборот на најверојатниот збор од множеството алтернативи може да се примени или композитниот модел кој ја дава сегментацијата која има најголема веројатност, или моделот кој ја максимизира a posteriori веројатноста на препознавањето т.е. сумата од сите сегментации. Во обата случаи, алгоритамот може да се организира за да се избегнат редундантните пресметувања, со примена на познатиот алгоритам на Viterbi.

Марковите вериги се покажале како моќно оружје за моделирање на фактот дека знаците (особено ракописните букви) немаат секогаш јасно дефинирани граници меѓу нив. Генерално, идеална дисекција не е можно да се постигне, но овој проблем се компензира со примена на Марковите вериги кои можат да се обучат со набљудување на сегментацијата на обучувачко множество примероци. Контекстот (фреквенциите на појава на знаци и зборови како и некои синтактички правила) може исто така да биде вклучен преку дефинирање на веројатностите на транзиција меѓу знаковните состојби.

Елементарните скриени Маркови модели кои опишуваат букви може да се комбинираат во повеќе model-discriminant Маркови модели за зборови или во единствен path-discriminant модел. Кај model-discriminant моделите по еден скриен Марков модел се конструира за секој различен збор кој системот треба да го препознава, додека кај path-discriminant моделот се креира еден единствен модел. Во првиот случај се испитуваат сите модели барајќи го оној кој најмногу одговара на множеството особини на зборот кој се препознава. Во вториот случај, препознавањето на зборот се одвива со пронаоѓање на најверојатните патеки низ единствениот модел, од кои секоја патека кореспондира на секвенца од букви. Лесно може да бидат конструирани path-discriminant модели и за големи лексикони, но тие генерално нудат помала прецизност од model-discriminant моделите.

Пронаоѓањето на најдобрата патека во скриениот Марков модел обично се изведува со примена на Viterbi алгоритамот, додека веројатностите на екстрахираните особини и транзициите меѓу состојбите може да бидат научени со помош на Baum-Welch алгоритамот.

Некои автори изложуваат методи за сегментација на знаци без примена на скриениот Марков модел. Кај повеќето од ваквите методи имплицитната сегментација се одвива паралелно со препознавањето, односно резултатите од препознавањето се користат за проценка на алтернативите и изведување на сегментацијата. Rocha и Pavlidis [RP93, RP95] претставуваат метода за препознавање на зборови репрезентирани како графови од особини. Системот ги споредува субграфовите од екстрахирани особини со предефинираните прототипови на знаци. Различните алтернативи се претставени со насочена мрежа чии јазли ги претставуваат препознатите субграфови. Препознавањето на зборот се изведува со барање на патека која дава најдобра интерпретација на особините на зборот. Некои автори предлагаат методи базирани на генерирање, тестирање и верификација на хипотези за сегментација и препознавање на знаци [HS86].

Кај холистичките пристапи зборовите се препознаваат како целина, без сегментација на знаци [Hon95]. Главен недостаток на оваа класа методи е што нивната примена е ограничена на одреден претходно дефиниран лексикон. Овој недостаток е особено изразен при обучувањето, имено неопходен е нов процес на тренирање на системот за препознавање при секоја промена или проширување на лексиконот од можни зборови. Оваа особина ги прави ваквите методи погодни за примена во апликациите каде лексиконот на можни зборови е статички дефиниран (непроменлив или ретко променлив) како препознавањето на чекови или поштенски броеви во адресите.

Wilkinson [Wil91, GWB+92] го испитува сегментирањето на знаци на масивно паралелни SIMD (Single Instruction Multiple Data) машини применувајќи хистограм и адаптивни статистички правила. Испитани се три различни алгоритми за сегментација на знаци имплементирани на масивно паралелна машина и споредени од аспект на точноста и брзината [WG92]. Garris [Gar92] изложува биолошки базиран систем за обработка на слика способен да изолира знаци од неструктуриран текст со симулација на организми применувајќи генетски алгоритми. Симулирани се вештачки организми со способност за визија, движење, конзумација, размножување и случајна мутација, чие интелигентно однесување вклучува пронаоѓање и "јадење" на знаците од текстот.

Скоро сите публикувани материјали кои го обработуваат проблемот на сегментација на знаци се концентрираат на сегментацијата на печатен или ракописен текст напишан на латиница, додека сегментирањето на текстови напишани на други писма ретко е обработувано. За сегментацијата на писмата кај кои знаците природно се допираат (арапското) вообичаено се користат пристапи слични на оние користени за сегментирање на ракописи. Кај ориенталните писма пак (кинеските, јапонските и корејските идеограми) проблемот на сегментација на знаци скоро и да не постои, бидејќи тие секогаш се пишуваат со пропорционален проред и значително растојание меѓу знаците.

Комплексноста на изведувањето прецизна сегментација зависи од природата на материјалот и неговиот квалитет. Генерално, ратата на грешки при сегментацијата расте доколку документите се слободно форматирани, со употреба на различни фонтови со различни големини, како и со опаѓањето на нивниот квалитет. Иако сегментацијата на знаците се разгледува одвоено, моменталните истражувања во областа на оптичкото препознавање на знаци сугерираат дека сегментацијата и препознавањето треба да се третираат интегрално за да се добијат поголеми точности особено во случајот на сложени (за препознавање) документи. Се смета дека токму мудрата комбинација на контекстот и степенот на доверливост (confidence level) на резултатите од препознавањето води кон значително подобрување на точноста во препознавањето на текстови.

4.5 Препознавање на знаци

Препознавањето на знаци е клучната компонента во секој систем за оптичко препознавање на знаци. За дадена слика на знак, алгоритамот за препознавање на знаци треба да одлучи за неговиот идентитет. Препознавањето на знаци претставува дел од областа на препознавањето на примероци.

4.5.1 Препознавање на примероци

Препознавањето на примероци (pattern recognition) е наука која се занимава со класификација (препознавање) или опис на објекти или настани. Класификацијата или описот обично се изведува врз база на некои одмероци или

особини земени од објектите или настаните, врз основа на некој модел. Според ова, препознавањето на примероци најчесто може да се подели во две фази:

- feature extraction – одвојување (екстракција) на некои особини од објектот или настанот; и
- classification – нивна класификација врз база на екстрахираните особини.

Главните цели на издвојувањето на особини се: намалување на димензионалноста на примерокот кој треба да се препознае и селектирање само на битните особини кои најдобро го карактеризираат, односно по кои се разликува од останантите примероци и занемарување на небитните особини. Изборот на особините за генералниот проблем на препознавање на примероци е повеќе умешност отколку наука. Сепак постојат основни теоретски водечки правци за избор на особините зависно од типот на проблемот – природата на примероците кои треба да се препознаат и својствата на класификаторот.

Постојат две главни групи на пристапи при препознавањето на примероци:

- статистички (decision-making) пристапи; и
- синтактичко-структурни (описни) пристапи.

Статистичките методи се decision-making пристапи кои секогаш донесуваат одлука за припадноста на примерокот кон некоја од класите или евентуално негово отфрлање доколку не можат да го препознаат. Структурните методи освен за класификација на примероците можат да се употребат и за нивен опис во поинаква форма.

Изборот на методата за класификација донекаде го диктира и изборот на техниката за издвојување на особини од примероците. Така на пример, статистичките методи за класификација типично работат со скаларни величини кои обично се организирани во вектор од особини (feature vector), додека структурните и синтактичките методи со структурни описни елементи, кои може да имаат најразлична претстава, а најчесто се претставени како низи од симболи, стебла, графови и сл. Постојат и хибридни пристапи за препознавање кои претставуваат комбинација на статистичките и синтактичко-структурните пристапи.

Главниот проблем кај статистичките методи за препознавање на примероци е изборот на особините кои ќе го репрезентираат објектот. Особините, земени како N одмероци од примероците, формираат N-димензионален векторски простор, во кој секој примерок може да се претстави како вектор чии координати кореспондираат на одмероците земени од него. Класификацијата претставува пресликување од овој N-димензионален векторски простор во просторот на класите кои треба да се препознаат. Заедничка карактеристика на статистичките методи е постоењето на база од дискриминантни функции кои го делат параметарскиот простор на повеќе региони, врз база на кои се врши класификацијата.

Методологиите кои се применуваат кај статистичките пристапи за препознавање на примероци, во основа ја имаат статистичката наука и можат да се поделат во три групи: параметарски и непараметарски методи и невронски мрежи кои се разгледуваат како посебна категорија. Кај параметарските методи функциите на густина на веројатност имаат математичка функционална форма

на некоја од познатите распределби (најчесто се користи гаусовата распределба), додека непараметарските методи го немаат ова ограничување и кај нив функциите на густина на веројатност можат да имаат произволна форма.

Најчесто користени статистички параметарски класификатори се: класификаторот според евклидово минимално растојание (Euclidean Minimum Distance – EMD), класификаторот според квадратно минимално растојание (Quadratic Minimum Distance – QMD) и нормалниот класификатор (NRML). Дискриминантните функции кај EMD се линеарни, додека кај QMD и NRML се квадратни. Најдобрата позната параметраска класификациона метода е базирана на Беаесовото правило. За примена на параметарските класификациони методи неопходно е познавање на генералната форма на функцијата на распределба на густината на веројатности за секоја класа, односно одредено *a priori* знаење, кое не е секогаш експлицитно, а понекогаш ни имплицитно достапно.

Непараметарските класификациони методи не се ограничени со структурна (параметарска) претстава на функциите на густина на веројатност. Тие вршат поделба на *N*-димензионалниот векторски простор со помош на хипер-рамнини на области на различните класи на примероци. Кај класификацијата по методата на најблизок сосед (Nearest Neighbour – NN) се креираат повеќе хипер-рамнини за раздвојување на класите и непознатиот примерок се класификува според најблиската до него класа. Погенерална верзија на методата на најблизок сосед е т.н. k-nerest neighbour (*k*-NN) класификациона процедура во која за припадноста на непознат примерок се одлучува врз база на гласовите од *k* негови најблиски соседи. Друга група на непараметарските класификациони методи применуваат адаптивни граници на одлучување и адаптивни дискриминантни функции, како и тежинско гласање при одлучувањето за припадноста на примерокот кон некоја од најблиските класи.

Постојат уште две главни групи на decision-making пристапи: кластер анализа и пристапи базирани на теоријата на непрецизирани множества (fuzzy sets). Кластерирањето претставува процес на групирање на примероците така да примероците во секоја група се слични меѓу себе, а групите се нарекуваат кластери. Кластер анализите можат да бидат разгледувани како посебно поле во методологиите за препознавање на примероци и може да се поделат на групирачки (agglomerative) и делбени (divisive) алгоритми. Како најважни меѓу групирачките алгоритми се сметаат алгоритмите за хиерархиско кластерирање бидејќи тие извршуваат постепено спојување на помалите кластери во поголеми со примена на различни критериуми за растојание помеѓу кластерите. За разлика од групирачките, делбените техники вршат постепена поделба на множеството примероци на сé помали и помали групи, со цел да се добие множество на кластери кои ја делат групата на подгрупи со слични примероци. Ваквите техники не само што ја опишуваат ситуацијата туку и наоѓаат соодветна репрезентација и врз основа на тоа го идентификуваат објектот.

Основната идеја на синтактичките пристапи за препознавање на примероци е во физичкото обликување на објектот по пат на рекурзивен опис на сложените форми со помош на поедноставни. Како што статистичките пристапи се темелат на статистиката и теоријата на веројатност, така синтактичките пристапи се потпираат на лингвистиката и регуларните граматики.

Регуларна граматика G претставува подредена четворка $G=(V_T, V_N, P, S)$ каде:

- V_T е множество на терминални симболи (примитиви);
- V_N е множество на нетерминални симболи (варијабли) кои се користат во процесот на генерирање;
- P е множество на продукции (rewriting rules) кои ги опишуваат правилата за замена;
- $S \in V_N$ е почетен симбол.

Вака генерираната граматка може да се употребува на два начина:

1. Генеративен – граматиката се користи за генерирање на низа од терминални симболи применувајќи ги продукциите од P.
2. Аналитички – за дадена низа од симболи (а) се оценува дали низата е генерирана со граматиката и ако е, (б) се бара нејзината структура (обично изразена преку секвенцата на применети продукции).

Регуларен јазик генериран со граматиката G, означен како $L(G)$, претставува множество на сите низи од симболи кои:

1. се состојат исклучиво од терминални симболи; и
2. секоја од нив е добиена од S со примена на продукциите P.

Класификацијата кај синтактичкото препознавање на знаци се сведува на анализа на влезната низа со цел да се одреди на која граматика и припаѓа, а ваквиот процес се нарекува синтаксна анализа или парсирање. Вака дефинираните граматики се еднодимензионални и не се соодветни за директна интерпретација на дводимензионални слики, па затоа се воведени поосодветни пристапи како јазикот за опис на слики (Picture Description Language – PDL), граматики за стебла (tree grammars), граматики за полиња (array grammars), plex граматики, граматики за графови (graph grammars) [Pav77] и др. Други структурни методи кои се применуваат за препознавање на примероци се верижните кодови дефинирани од Freeman како и подобрените проширени кодови – IEF.

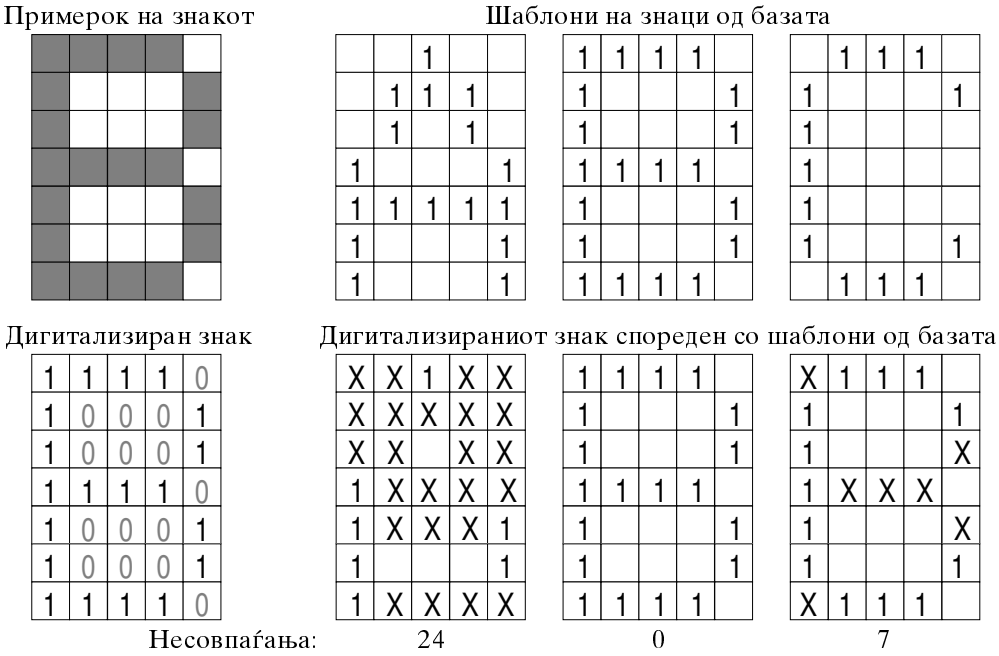
Хибридните методи се обидуваат да го обединат синтактичкиот и статистичкиот пристап, на пример со примена на стохастички регуларни јазици или разгледувањето на повеќето граматики и односите меѓу нив како кластери. Во последно време забележителна е сé почестата примена на невронските мрежи за решавање на проблемите на препознавање на примероци. Иако во основа претставуваат статистички пристап за препознавање на примероци, невронските мрежи се разгледуваат како посебна класа методи. Специфичните карактеристики на невронските мрежи:

- паралелизам
- учење од примероци
- генерализација

како и нивната структурна организација која е изведена од начинот на организација и функционирање на човековиот мозок кој и понатаму е неприкосновен во задачите на препознавање и класификација, ги прави идеални кандидати за примена во полето на препознавањето на примероци.

4.5.2 Препознавање на печатени знаци

Препознавањето на печатени знаци спаѓа во класата на проблеми кои опфаќаат препознавање на визуелни информации, поточно статички визуелни информации односно слики. Препознавањето на сликовни информации (pictorial pattern recognition) претставува идентификација на формите присутни на сликата која најчесто е претставена како дводимензионална матрица. Во процесот на препознавање на печатени знаци се врши асоцирање на сликата на изолираниот знак со неговата симболичка претстава (ASCII код на пр.), односно негово распоредување во некоја од дисјункните класи на знаци кои треба да се препознаат.



Слика 4.15 Споредување на шаблони (template matching)

Процесот на препознавањето на печатени знаци, како и повеќето пристапи за препознавање на примероци најчесто опфаќа две фази: одвојување (екстракција) на некои особини од сликата на знакот и класификација на знакот врз база на екстрахираните особини. Постојат и методи кои вршат директна класификација на самите слики на знаците, по пат на нивно споредување со шаблони (template matching). Кај ваквите методи за особини практично се земаат самите вредности на пикселите, без некоја претходна трансформација над нив. Споредувањето со шаблони е традиционална метода која се користела кај првите системи за препознавање на знаци. Таа се состои во споредување на дигитализираниот знак претставен како бинарна матрица со множество на прототипови на знаци (шаблони) сместени во меморијата. Кога примерокот ќе се совпадне (или приближно ќе се совпадне) со маската (шаблонот) на некој од знаците во базата, му се доделува симболичката претстава (ASCII код) на знакот со чиј шаблон се совпаднал (Слика 4.15). Ова се повторува за секој знак, а методите за споредување можат да бидат со најразлична комплексност, од едноставни, како еден-спрема-еден споредување, до комплексни како стебла за одлучување во кои се споредуваат само одредени пиксели [Sha95, JKN+96], или се врши соодветно поместување на примерокот заради негово подобро совпаѓање

со шаблонот. Овие техники се осетливи на шум и различни типови на писмо, односно ги препознаваат само оние писма чии прототипови се сместени во базата.

Одвојување на особини (feature extraction)

Процесот на одвојување на особини од сликата на знакот се нарекува уште и анализа на облици (shape analysis). Методите за анализа на облици може да се поделат според повеќе критериуми [Pav78]. Зависно од тоа дали ја испитуваат само границата меѓу објектот и позадината или целата слика, се делат на *екстерни* и *интерни*. Исто така, оригиналната слика може да биде претставена преку *скаларни големини* или со *структурни описни елементи*. Зависно од тоа дали од издвоените особини може повторно да се реконструира појдовната слика или не, се делат на методи кои ги сочувуваат информациите (*information preserving*) и оние кои не ги сочувуваат (*information nonpreserving*).

Поделбата на методите за екстракција на особини, може да се направи и од аспект на техники во *просторен домен* и техники на *скаларна трансформација*. Техниките кои оперираат во просторен домен ја трансформираат сликата во друга слика, а не во поле од скаларни големини. Од аспект на понатамошната обработка, скаларните трансформациони техники се посоодветни за добивање на влезните големини за статистичките класификатори, додека просторно доменските техники се поосодветни за генерирање на влезот за синтактичките и структурните класификатори. Информациониот домен од кој се екстрахираат особините, може да биде просторниот домен или трансформациониот домен на некоја трансформација (фреквентниот домен на пр.).

Особините најчесто се екстрахираат со примена на некој алгоритам или оператор за екстракција на особини над сликата. Тие можат да бидат нисконивовски (*low-level*) како вредностите на пикселите, нивната густина и распределба или високонивовски (*high-level*) како меѓусебниот распоред и однос на основните градбени објекти (точки, линии, криви) на сликата. Pavlidis [Pav93] особините за опис на формите ги дели на *структурни* и *инцидентни*, односно методи за *структурна анализа* и методи за *глобална анализа*. Пример за структурни особини се: потези (strokes), конкавност на контури на региони, отвори и сл., додека инцидентни особини се: битовите маски, фуриеовите коефициенти и сл.

Една од најрано користените техники за екстракција на особини е методата на геометриски моменти, дефинирани како:

$$m(u,v) = \iint f(x,y)x^u y^v dxdy \tag{4.14}$$

Методата на моменти е потполно information preserving метода и може да се класификува како интерна техника со скаларна трансформација. Наместо директно користење на моментите често како особини се користат одредени линеарни комбинации од нив, кои се инваријантни во однос на одредени трансформации. Ова води кон поопштата дефиниција на моментите како интеграл од производот на сликата со секој од множеството полиноми дефинирани над 2D регионот:

$$m(u,v) = \iint f(x,y)h_{uv}(x,y)dxdy \tag{4.15}$$

каде $f(x,y)$ е бинарната слика, а $h_{uv}(x,y)$ е полином од степен u по x и v по y. За наједноставниот случај кога $h_{uv}(x,y)=x^u y^v$ се добиваат геометриските моменти. За екстракција на особини по пат на моменти за потребите за препознавање знаци најчесто се користат полиномите на: Legendre, Zernike и pseudo-Zernike. Bailey и Srinath [BS96] предлагаат ортогонални моменти инваријантни во однос на положбата, големината и ротацијата на знакот.

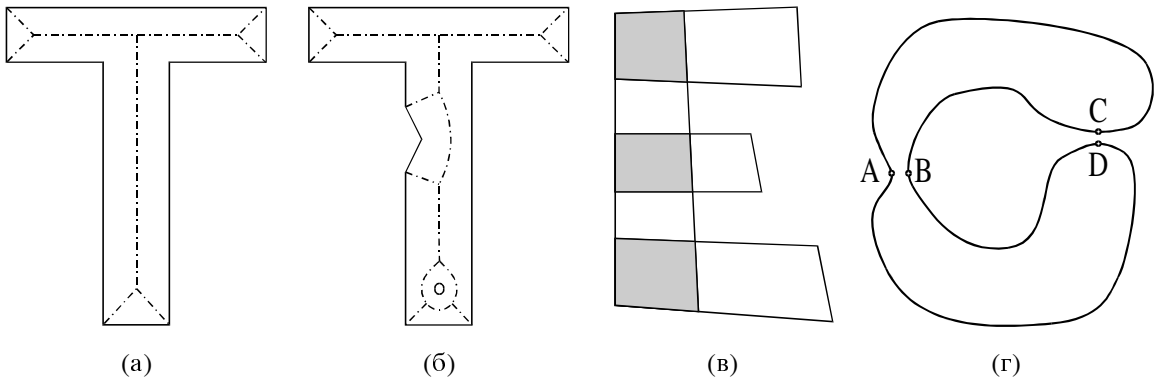
Други често користени методи со скаларна трансформација се техниките за екстракција на коефициенти од трансформационите домени на некоја од трансформациите како: Fourier, Hadamard, Rapid, Karhunen-Loève, Walsh и др. Најчесто овие трансформации се користат како дводимензионални и оперираат над целата слика (како интерни методи), но некои од нив можат да се применат и над веќе трасираната граница меѓу објектите и позадината (како екстерни методи) или скелетот на објектите добиен со некоја просторна трансформација. Ваквите трансформации даваат множество коефициенти од кои се избираат неколку како репрезенти на објектот. Иако овие методи во основа се information preserving методи, од бројот на издвоените коефициенти зависи колкаво количество од информацијата ќе биде сочувано. Според ова можеби би било посоодветно поделбата на методите да се направи на оние кај кои загубата на информација може да се контролира и оние кај кои не може да се контролира.

Меѓу најупотребуваните интерни просторно-доменски техники е трансформацијата на средишни оски (median axis transformation – MAT) со која "целата" слика се трансформира во линиски цртеж кој се нарекува скелет (Слика 4.16 а). Средишните оски се дефинираат на следниов начин: Ако S е множество соседни точки во рамнина чија граница е B, тогаш за секоја точка $X \in S$ може да се најде најблиската до неа соседна точка $Y \in B$. Доколку точката X има повеќе од една вакви соседни точки, тогаш таа припаѓа на средишната оска на S. Вака добиениот скелет од средишни оски ја сочувува базичната топологија на објектот и може да се користи за одвојување на информации за формата на оригиналната фигура или за нејзина класификација со некој од структурните класификатори. Доколку за секоја од точките на средишните оски се чува и нејзиното растојание до најблиската граница, можна е потполна реконструкција на појдовната слика. При препознавање на знаци ова ретко се прави, односно се разгледува само костурот со што неконтролирано се губат информации. Пронаоѓањето на средишните оски е релативно комплексна операција. Лошата страна на овој тип техники е што се многу осетливи на шум, така што произволно мал отвор во објектот или негова деформација може да произведе значителна промена на неговиот костур (Слика 4.16 б). За потребите на препознавање на знаци почесто се употребуваат техники за т.н. тенчење (*thining tehniques*) [LS95] кои вршат итеративно бришење на граничните пиксели сé додека со нивното отстранување не се нарушува поврзаноста на објектот [Dav90] (Слика 4.17 б,в).

Друга група на декомпозициони техники се базираат на разложување на објектот. Ова може да биде разложување на потези (основните линии кои се употребуваат при пишувањето на буквите) или на множество конвексни полигони (primary convex subsets – PCS) (Слика 4.16 в).

Најпознати екстерни просторно-доменски техники се техниките за кодирање на границата на објектот, на пример со Freeman-овиот верижен код. Главните предности на овие техники е нивната брзина, малите потреби за мемориски ресурси и расположливоста на развиени генерални методологии за

нивна обработка како теоријата на формалните јазици. Главниот недостаток им е што геометриски блиските точки на сликата можат да се појават во кодираната низа доста далеку една од друга (Слика 4.16 г). Слични техники кои ја анализираат границата на објектот се техниките кои вршат хиерархиско преклопување на границата со параметарски криви (splines), и анализа на нивната закривеност заради опис на контурите на објектот и детекција на агли и страни.



(а) (б) (в) (г)

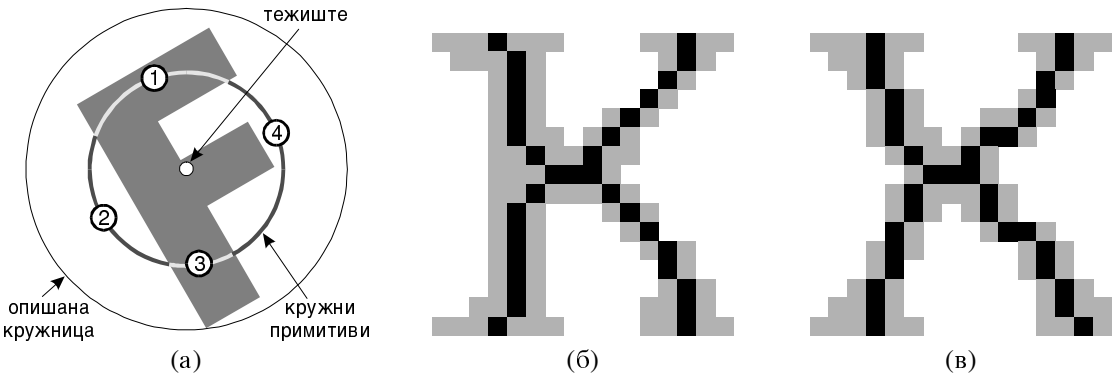
Слика 4.16 (а,б) Примери за трансформација на средишни оски (MAT); (в) Пример за декомпозиција во множество на примарни конвексни полигони (PCS); (г) Илустрација на проблемот со верижно кодирање на границата: иако просторно точката A е блиску до B и C е блиску до D, тие се релативно далеку по должината на границата

Други особини кои често се користат при препознавањето на знаци се целуларните (ќелиски) особини (cellular fatures). Со оваа метода матрицата со сликата на знакот се дели на $n \times n$ компоненти (ќелии). Во секоја од овие ќелии се испитува постоењето на хоризонтални, вертикални или дијагонални потези кои минуваат низ неа, кои ако постојат, заедно со позицијата на ќелијата се бележат како особини. Ваквите особини потоа можат да бидат споредени со особините од базата со шаблони. Слично се добиваат и т.н. мрежни особини (mesh features) кои се добиваат со пресметување на пропорцијата на црните и белите површини (потези – позадина) во секоја од компонентите.

Проекциите освен за сегментација на знаците можат да се користат и како особини при нивното препознавање и најчесто се користат за груба предкласификација на знаците во одреден број дисјунктни групи или во комбинација со други особини. Најчесто се користат хоризонталната, вертикалната и дијагоналните (коси) проекции [Pav84] или нивните фуриеови трансформации, како и разни верзии на регионални проекции кои се применуваат на делови од сликата на знакот поодделно. Tang и Suen [TS92] ја применуваат регионалната проекциона трансформација (regional projection transformation – RPT) поточно нејзината DDRPT верзија, погодна за паралелно извршување за добивање на поврзан објект без отвори кој понатаму се користи при препознавањето на кинески знаци.

Menu *et al.* [MLL92] за препознавање на печатени знаци употребуваат циркуларни (кружни) примитиви кои се инваријантни во однос на транслацијата, ротацијата и скалирањето. Инваријантноста во однос на транслацијата се обезбедува со поставување на центарот на кружницата за екстракција на примитиви во тежиштето на знакот. Радиусот на оваа кружница е пропорционален на радиусот на опишаната кружница околу знакот, за да се обезбеди инваријантност во однос на скалирањето (големината на знакот).

Примитивите се избираат како аглите меѓу последователните точки од кружницата во кои таа влегува или го напушта објектот (Слика 4.17 а), со што се постига инваријантноста во однос на ротацијата. Класификацијата се изведува со помош на двостепена невронска мрежа, од која првиот степен врши класификација на знакот во 9 суперкласи од слични знаци од каде второто ниво одлучува за конечната припадност на знакот, користејќи при тоа дополнителни примитиви.



Слика 4.17 (а) Кружни примитиви; (б,в) Различни знаци кои при тенчењето резултираат во слични костури

Shustorovich [Shu94] и Garris *et al.* [GWW91] користат дводимензионална Gabor трансформација за екстракција на особини од сликите со знаци, кои се користат при нивната класификација со помош на стандардна невронска мрежа и local receptive field мрежа. Lee [Lee96] за екстракција на особини ги користи маските на Kirsch за потенцирање на потези во одредени правци, применети над делови од сликата на знакот, а класификацијата ја врши со повеќенивовска кластерска невронска мрежа.

Kahan *et al.* [KPB87] за препознавање на знаци од произволен фонт и големина првин ги екстрахира клучните потези на знакот, по пат на тенчење со помош на line adjacency graph (LAG) претставата на сликата на знакот. Секој потег се претставува со четири параметри кои го карактеризираат (координатите на неговиот центар, должината и правецот). Класификацијата се одвива на неколку нивоа, и тоа прво врз база на димензиите и положбата на знакот во редот со текст, потоа според кодираните потези од кој е формиран, а во случаите каде ова не е доволно (Слика 4.17 б,в) се испитува и контурата на знакот.

Rocha и Pavlidis [RP94] ги претставуваат знаците како графови од особини, а класификацијата се спроведува со нивна споредба со прототипови на знаците. Прототиповите се споредуваат со хомеоморфни субграфови изведени од сликата на зборот, со што е овозможено препознавање и без претходна сегментација на зборовите до изолирани знаци [RP93]. Овозможено е и пресметување на цена на потребна трансформација над графот, за тој да се совпадне со прототипот, односно добивање мерка на сличност.

Baird [Bai88] применува хибриден статистичко-синтактички пристап за препознавање на печатени знаци, базиран на процедурата наречена идентификација на особини (feature identification). Од скелетот на знакот се одвојува множество на неподредени структурни особини како потези и отвори, кои потоа се пресликуваат во статистичка (скаларна) форма во вид на подреден

бинарен вектор, кој претставува точка во параметарскиот простор. Тренирањето на класификаторот се изведува по Баесовата техника, со која се врши кластерирање на ваквите точки заради идентификација на регионите од интерес во параметарскиот простор, од кои секој се асоцира кон соодветен бит од бинарниот вектор со особини.

Класификација

Во основа методите за класификација може да се поделат на два вида: адаптивни методи базирани на учење (обучување) и методи базирани на правила. Адаптивните методи базирани на учење може понатаму да се поделат на: надгледувано учење (обучување од страна на учител – supervised learning) и самообучување (самоорганизација – self-organization). Во системите за оптичко препознавање на знаци за класификација најчесто се употребуваат: rule-based методи (базирани на правила), image-based методи (базирани на слика), статистички методи и невронски мрежи.

Kaj методите базирани на правила, програмата за класификација се конструира да ги разликува различните знаци преку пишување на правила кои треба да ги следи при одлучувањето. Ваквите правила експлицитно се наведуваат во системот (се програмираат) во вид на математички формули. Методите за класификација базирани на правила за одлучување обично се среќаваат кај системите кои класификацијата ја изведуваат врз база на издвоени структурни особини.

Адаптивните методи базирани на учење се обидуваат да симулираат интелигентно однесување преку адаптивно учење. Нивната поделба на надгледувано учење и самообучување, од аспект на препознавањето на знаци, се однесува само на екстракцијата на особините кои се користат при класификацијата, а не и на класификацијата во финалните класи чиј број е определен и нема врска со методите кои вршат автоматска кластеризација. Така на методите со надгледувано учење им се нудат веќе на некој начин филтрирани и преобразени информации (екстрахирани особини од сликата на знакот) врз база на кои треба да ја извршат класификацијата. На методите со самообучување им се проследуваат сирови информации (најчесто самата слика на знакот), и од нив се очекува самите да го "научат" издвојувањето на репрезентативните и игнорирање на небитните особини.

Методите за класификација базирани на слика всушност ги опфаќаат template matching методите кои вршат директна класификација на самите слики на знаците. Статистички параметарски класификатор кој најчесто се користи при препознавањето на знаци е Баесовиот класификатор, а се среќаваат и EMD, QMD и NRML класификаторите [BCG+93]. Од непараметарските методи најчесто користена е методата на најблизок сосед во различни варијанти: NN, *k*-NN и WSNN – Weighted Several NN [BCG+93], како и различни применети метрики за добивање на растојанието од прототипот до примерокот [SCD94, SMK+94, Waa95, Por95, Nes95] и оптимизации за нивна имплементација [GCB95, HT96, Nes95a] и обучување [SDH+95, RBT+95].

Во последниве години за препознавање на знаци сé почесто како класификатор се употребуваат невронските мрежи. Перформансите кои се во најмала рака еднакви, а често и подобри од најдобрите класични статистички

методи, како и едноставната реализација и обучување ги прави невронските мрежи одлични кандидати на кои сé почесто се доверува задачата на класификација во системите за оптичко препознавање на знаци. Се среќаваат најразлични видови и архитектури на невронски мрежи како: Multi-Layer Perceptron, Neocongitron [LDT95], каскадни и повеќестепени невронски мрежи [MLL92, JS92], Radial Basis Function NN (RBF), веројатносни невронски мрежи (Probabilistic NN) [RTT96], Хопфилдови невронски мрежи (Hopfild NN) [GAP91], невронски мрежи со рекурентна пропагација на грешка (Recurrent Error Propagation Networks) [SF92, Sen92, Sen94] и др.

И статистичките методи и невронските мрежи како адаптивни методи се базирани на учење, односно за нивно обучување е потребно обучувачко множество со примероци. Статистичките методи од обучувачкото множество се обидуваат да ги извлечат статистичките законитости, врз база на кои потоа ќе ја вршат класификацијата на непознатите примероци. Параметарските методи вршат нагодување на параметрите на веќе однапред усвоена функција на распределба на густината на веројатност со цел таа што подобро да се преклопи со фактичката распределба изведена од потребно обучувачко множество. Нивната лоша страна е што формата на усвоената функција на распределба можеби нема да може да обезбеди доволно добра репрезентација на вистинската распределба. Кај непараметарските методи функцијата на распределба на густината на веројатност не е однапред дефинирана и може да има потполно произволна форма во зависност од примероците во обучувачкото множество. Проблемот кај непараметарските методи е што бројот на параметри потребни за опишување на функцијата на распределба на густината на веројатностите може да биде доста голем и да расте со димензионалноста на проблемот. Постојат и т.н. семипараметарски методи кои дозволуваат употреба на доста широка класа на функционални форми за кои бројот на адаптивни параметри може систематски да се зголемува со цел изградба на пофлексибилни модели кај кои бројот на адаптивни параметри може да се контролира независно од димензионалноста на проблемот. Невронските мрежи може да се сметаат за еден вид семипараметарски статистички методи.

Невронска мрежа како класификатор

Невронските мрежи или поточно вештачките невронски мрежи претставуваат масивно паралелни, слабо сериски, масовно поврзана мрежа од релативно едноставни пресметувачки елементи наречени *неврони*. Нивните главни предности се: масивниот паралелизам кој се должи на кооперативната и натпреварувачка работа на невроните, ограничената отпорност на шум и грешки (fault-tolerance), како и откривањето на елегантни алгоритми за обучување на одредени класи невронски мрежи.

Влезот во неврон со n влезови може да се претстави како примерок во форма на вектор $\mathbf{X} \in \Re^n$, или во случај на бинарен вектор како $\mathbf{X} \in [0,1]^n$. Секој неврон пресметува релативно едноставна функција од неговите влезови и ја проследува до другите неврони со кои е поврзан преку своите излези. Во пракса се среќаваат различни функции, но најчесто употребувани се: линеарната, функцијата на праг и сигмоидалната функција. Секој неврон содржи множество параметри кои може да се менуваат во процесот на учење. Најчесто користените параметри се т.н. тежини претставени како n-димензионален тежински вектор

$\mathbf{W}_i \in \Re^n$. Секој неврон го пресметува скаларниот производ $\mathbf{W}_i \cdot \mathbf{X}$, кој потоа се пропушта низ функцијата на невронот и како ваков се проследува на излезот. Обично еден од влезовите се држи константен, а на него соодветната тежина се нарекува праг. Можностите за претстава и пресметка зависат од функциите кои ги пресметуваат секој од невроните како и од архитектурата на мрежата – бројот на невроните и начинот на кој се поврзани. Невронските мрежи организирани во слоеви кај кои сигналите се простираат во една насока (layered feed-forward network) се најчесто применуваната архитектура во апликациите за препознавање на примероци.

Повеќеслојниот перцептрон (MLP) е невронска мрежа во која невроните се организирани во повеќе слоеви. Излезите од невроните од секој слој (освен последниот) се поврзани на влезовите на секој од невроните од наредниот слој. Врски помеѓу невроните од ист слој и од несоседни слоеви не постојат. Повеќеслојниот перцептрон е во пракса најчесто употребуваната архитектура на невронска мрежа, што се должи најмногу на едноставната имплементација и елегантниот back-propagation алгоритам за нејзино обучување.

Учењето најопшто може да се дефинира како процес на стекнување нови знаења, организација на стекнатите знаења во применлива репрезентација, развој на перцептуални, моторни или спознајни способности и откривање на нови факти, хипотези или теории за светот преку истражување, експериментирање, индукција, дедукција или абдукција [Hon94]. Во областа на машинската интелигенција односно машинското учење, тоа може да се дефинира како процес со кој системот ги подобрува своите перформанси во решавањето на доверените задачи стекнувајќи искуство со тек на времето. Најголем дел од истражувањата за обуката на невронските мрежи досега беа фокусирани кон алгоритми кои вршат промена на променливите параметри (тежините) на невроните во невронските мрежи со одредена однапред усвоена архитектура, употребувајќи множество од познати вредности на функцијата кој треба да се апроксимира или примероци кои треба да се класифицираат. Обучувањето на невронските мрежи се темели на математички формализам кој настојува да ја минимизира грешката при класификацијата или апроксимацијата. Задачата на алгоритамот за учење е да пронајде множество на тежини кои водат кон задоволителна апроксимација на непознатата функција за дадените примероци, што во основа претставува оптимизационен проблем, при што како мерка за успешноста на апроксимацијата најчесто се користи средната квадратна грешка меѓу бараните (посакуваните) и актуелните излези од мрежата.

Не постои начин за ad hoc избор на архитектура која е соодветна за решавање на одреден проблем, па секогаш постои можност да се избере несоодветна архитектура која нема да може задоволително да одговори на проблемот или пак ќе биде предимензионирана. За избегнување на овој проблем, во последниве години се вршат опсежни истражувања во изнаоѓањето алгоритми кои овозможуваат и менување на архитектурата на невронската мрежа во процесот на нејзиното обучување. Техниките на кастрење (pruning) [TF95, TF96] со кои се отстрануваат редундантните и непотребни неврони и врски меѓу невроните, како и техниките за динамичко растење на невронските мрежи, исто така отвараат нови можности и области за истражување.

Во зависност од применетата невронска мрежа, се применуваат различни алгоритми за обучување, оптимизација на обучувањето како и оптимизација во

изведбата на самите невронски мрежи. Кај класичните повеќеслојни невронски мрежи (MLP) обучувањето најчесто се одвива според back-propagation алгоритамот. Примената на алгоритми произлезени од back-propagation алгоритамот кои аплицираат променлива рата на учење [Jac88, JF93, MF95, FKS96], соодветна иницијализација на тежините [TF97], како и соодветна подготовка и проверка на примероците од множеството за обука [TR93] можат значително да ја забрзаат конвергенцијата на невронската мрежа при тренирањето.

Radial basis functions невронските мрежи поради своите добри својства при отфрлањето на сомнителни примероци и побрзото обучување исто така често се среќаваат во апликациите на препознавање на примероци. Кај овие двослојни мрежи, активацијата на скриените неврони се одредува според растојанието помеѓу влезниот вектор на примерокот и векторот на прототипот. Обучувањето на radial basis functions невронските мрежи се одвива во две фази: во првата фаза се нагодуваат параметрите на базните функции (скриениот слој на неврони) со примена на ненадгледувано учење (без учител), додека во втората фаза се одредуваат тежините на невроните од излезното ниво што бара решавање на линеарен проблем. Gentric и Withagen [GW93] предлагаат метода за забрзување на евалуацијата кај ваквите невронски мрежи со помош на стебло.

Нешто поретко се среќаваат имплементации на веројатносни невронски мрежи (PNN), мрежи со повратна пропагација на грешка [SF92, Sen94, SR95] и мрежи со променлива архитектура [LTS+92, AP93]. Веројатносните невронски мрежи претставуваат имплементација на статистичкиот баесов класификатор во вид на невронска мрежа [Spe88]. PNN мрежите ги користат податоците од обучувачкото множество за да ги одредат параметрите на функциите на распределба кои се користат при одредувањето на веројатноста одреден примерок да е припадник на одредена класа. Опционо оваа веројатност може да се комбинира со a priori веројатноста (релативната фреквенција) на секоја од класите, за да се одреди најверојатната класа за даден примерок. Добрите страни на PNN се што се обучуваат многу побрзо од мрежите кои се обучуваат според back-propagation алгоритамот и што параметарот кој влијае на генерализацијата може да се нагоди без повторно тренирање на мрежата. Лошата страни на PNN е што за добра генерализација се потребни голем број на обучувачки примероци, од кои секој претставен како посебен неврон во скриениот слој ја прави мрежата комплексна, и доколку не е посебно нормализирана и оптимизирана, побавна при евалуацијата од класичниот MLP. Невронските мрежи со повратна пропагација на грешка се специфични по тоа што дел од невроните од излезниот слој се поврзани со повратни врски на влезовите од невроните во влезниот слој. Ваквата архитектура обезбедува одредена динамика при евалуацијата и се обучува по методи слични на обуката на скриените маркови модели. Невронските мрежи со променлива архитектура немаат фиксна структура и топологија, туку таа се менува според некои законитости во зависност од тренирачките примероци и во основа оваа промена на структурата може да се прави во текот на самото обучување на мрежата или по завршувањето на обучувањето на одредена структура со отстранување на редундантните и непотребни неврони и врски меѓу нив.

Сао *et al.* [CAS93] за препознавање на печатени знаци применуваат невронски мрежи во каскадна врска, при што првата се употребува за

екстракција на особини од сликата на знакот, а втората за самата класификација. Jouny и Sheridan [JS92] употребуваат двостепена повеќеслојна невронска мрежа за препознавање на печатени знаци. Првиот степен е сочинет од две одвоени мрежи од кои секоја е обучена да препознава само дел од знакот (горната и долната половина), чии излези се презентираат на друга мрежа (вториот степен) која ја врши финалната класификација. Zhou и Pavlidis [ZP93, ZP94] предлагаат реализација на класификатор во повеќе хиерархиски нивоа, кој тргнувајќи од најгенералната поделба, на секое наредно ниво врши вклучување на нови особини при класификацијата која е посебно оптимизирана за решавање на соодветниот подпроблем. Baird и Nagy [BN94] ги испитуваат можностите за динамичко обучување на класификаторите во текот на препознавањето и нивно автоматско прилагодување кон формата на знаците кои се препознаваат.

Проценка (евалуација) на класификаторите

Од аспект на резултатот на класификацијата можни се два исхода: примерокот да биде препознат и распореден во одредена класа или да не биде препознат – да не може да се распореди во ниту една од класите, да биде отфрлен, односно класификаторот да **одбие** да го распореди во некоја од класите. Од аспект на одлуката при класификацијата, класификаторите може да се поделат на hard-decision и soft-decision класификатори. Hard-decision класификаторите се "строги" и носат цврста одлука за припадноста на примерокот на една и само една точно одредена класа од можните класи. За ваквите класификатори примерокот или е или не е припадник на дадена класа. Soft-decision класификаторите се "поблаги" и при изразувањето на резултатот од класификацијата употребуваат fuzzy логика, односно даваат мерки на верување за припадноста на примерокот кон секоја од класите. Проблемот на оптичко препознавање на изолирани знаци во основа бара цврста одлука за припадноста на примерокот бидејќи тој сигурно претставува еден и само еден знак и не може истовремено да претставува друг знак, па дури и ако личи на него. Претставата пак која што ја даваат soft-decision класификаторите може корисно да се употреби при контекстното постпроцесирање и врз база на контекстот да се донесе конечната одлука.

Конечната одлука која ќе се донесе за даден знак може да биде или точна (знакот да биде правилно препознат) или неточна (знакот да биде препознат како некој друг). Според ова, од аспект на правилноста при класификација на еден примерок можни се три исходи:

- примерокот е правилно препознат – донесена е правилна одлука;
- примерокот е погрешно препознат (како некој друг) – донесена е погрешна одлука;
- примерокот не е препознат – не е донесена никаква одлука.

Причините поради кои доаѓа до грешки во класификацијата кај оптичко препознавање на знаци Pavlidis [Pav93] ги дели во пет групи:

1. Грешки поради сличност на знаците (c-e-o, C-O-0, 5-S, 8-B);
2. Грешки поради деформација на печатениот материјал – слаб квалитет на печатениот материјал кој доведува до деформација на знаците, губење на нивни делови или појава на дамки од мастило кои ја деформираат формата на знаците;

3. Грешки поради дигитализацијата, што во основа претставува неповратно губење на дел од информацијата;
4. Грешки поради несоодветен избор на особини кои ќе бидат користени при препознавањето на знаците;
5. Грешки поради несоодветен дизајн на класификаторот, најчесто поради несоодветно обучувачко множество или избор на класификатор кој не може да одговори на комплексноста на проблемот.

Причината за најголемиот број на грешки кај системите за оптичко препознавање на знаци – неправилната сегментација, според оваа поделба практично е опфатена со причините од втората и третата група. Зависно од квалитетот на сликата на документот перформансите на одредени класификатори може значително да се разликуваат. Дисторзиите и дефектите кои настануваат на сликата, од аспект на нивното влијание на оптичкото препознавање на знаци е обработувано од повеќе истражувачи [Bai93, JKN93, WGJ92].

За правилна проценка и избор на соодветна метода за решавање на одреден проблем често пати е потребно воведување на соодветна квантитативна мерка за мерење на перформансите на различните методологии. Ова најчесто се прави по пат на споредување на набљудуваната променлива (излезот од OCR систем) со референтната променлива (вистинската содржина внесена или проверена од човек) под контролирани услови. Успешноста на класификаторите најчесто се изразува преку:

- рата на (правилно) препознавање (recognition rate) – R_{rec};
- рата на грешка (error/substitution rate) – R_{err};
- рата на отфрлање (rejection rate) – R_{rej}

дефинирани како:

$$R_{rec} = \frac{c}{N}; \qquad R_{err} = \frac{e}{N}; \qquad R_{rej} = \frac{r}{N} \tag{4.16}$$

каде:

N е вкупниот број на знаци пуштени на препознавање;
c е бројот на правилно препознати знаци;
e е бројот на погрешно препознати знаци; и
r е бројот на непрепознати (отфрлени) знаци.

Грешката на погрешно препознавање, најчесто останува неоткриена без пообемно ангажирање на човеков труд, со што целата поставеност на системот за автоматско оптичко препознавање на знаци доаѓа во прашање. Од друга страна отфрлените знаци и не претставуваат грешка, но факт е дека системот не успеал автоматски да ги препознае, што значи и неопходно ангажирање на човековото внимание за исправање на овие од системот посочени недоумици при препознавањето. Зависно од барањата на системот, овие две рати на грешки не мораат да имаат подеднакво значење. Со соодветно нагодување на параметрите на класификаторите можно е донекаде да се влијае на односот на R_{rec}, R_{err} и R_{rej} т.е. да се намали R_{err} за сметка на зголемена R_{rej}, но и нешто намалена R_{rec} и обратно. Така на пример, за банкарски апликации соодветен би бил класификатор со што помала R_{err} (по можност нула), а со дозволена R_{rej} со

прифатлива вредност. За апликациите кои вршат препознавање на текст само за негово индексирање заради ефикасно пребарување, кои работат со огромни количини на податоци, ангажирањето на човекот треба да се сведе на минимум, односно пријавите на непрепознати знаци не би се сервисирале, па пожелно е да ги има што помалку, за сметка на грешките при препознавањето кои може да се толерираат до одредена мера до која не влијаат битно на финалните можности за пребарување [TBC+94].

Перформансите на класификаторите може да се разгледуваат од повеќе аспекти:

- потребните мемориски ресурси при обучувањето;
- потребните процесорски ресурси при обучувањето (брзина на обучување);
- потребните мемориски ресурси при нивната работа (препознавање);
- потребните процесорски ресурси при нивната работа (брзина на препознавање);
- способноста за генерализација;
- отпорноста на шум и деформации;
- точноста на препознавањето, односно R_{rec}, R_{err} и R_{rej}, и сл.

За намалување на потребните мемориски а делумно и процесорски ресурси кај MLP невронските мрежи може да се примени дискретизација на тежините. Подобрување на генерализирачките способности на класификаторите најлесно се постига со презентирање на што поголем број различни примероци при обучувањето. Во случаите кога се располага со ограничен број на примероци може со додавање на шум и контролирано изобличување да се генерираат поголем број вештачки примероци, со што се подобрува генерализацијата и отпорноста на шум на класификаторите [ADG95]. За проценка на перформансите на класификаторите (cross-validation), а кај некои алгоритми и во процесот на обука (за проценување на моментот кога треба да се престане со обучувањето) се користи множество за тестирање (проценка). Тоа е составено од примероци кои не учествуваат во процесот на обуката и се користат за проценка на препознавачките можности на класификаторот на примероци кои не му биле презентирани. Преглед и споредба на перформансите на неколку класификатори имплементирани во системи за оптичко препознавање на знаци е даден во [Wil92, BCG+93, BCG+94, KJ94].

Комбинирање на класификатори

За подобрување на перформансите (точноста и генерализацијата) во класификаторските проблеми некогаш се користат и повеќе класификатори, што е наречено *комбинирање на класификатори*, односно употреба на комитети или ансамбли од класификатори. Со комбинирањето на повеќе класификатори кои конкурентно и кооперативно одлучуваат можно е да се постигне точност поголема од точноста на најдобриот класификатор во групата. При комбинирањето на класификаторите се наметнуваат две прашања: колку и какви класификатори да се употребат и на кој начин да се комбинираат нивните резултати за да се добие подобар резултат.

Што се однесува до типот на класификаторите, можно е комбинирање на еднородни или разнородни класификатори, а што се однесува до нивниот број,

колку повеќе класификатори се комбинираат толку повеќе се подобруваат резултатите, но тие зависат и од други параметри како начинот на нивното комбинирање и степенот на зависност (корелираност) меѓу класификаторите. Некорелираноста на класификаторите е клучна за подобрување на перформансите при нивното комбинирање. Комбинацијата од класификатори чии одлуки се доста корелирани не донесува никакво подобрување во препознавањето без оглед на начинот на нивното комбинирање, односно перформансите на ваквата комбинација не можат да бидат подобри од перформансите на најдобриот класификатор во групата. Границите на подобрувањето на перформансите на комбинацијата во зависност од перформансите на поедните класификатори е испитувано од повеќе истражувачи [Ali96, Mat96, Tum96, TG96, TG96a]. За намалување на корелираноста на класификаторите предвидени за комбинирање, предложени се повеќе методи за нивно обучување од кои повеќето опфаќаат начини на поделба на обучувачкото множество на одреден број подмножества од кои при обуката на одредени класификатори се презентираат само одредени подмножества (k–1-of-k training) или генерирање дополнителни примероци за обука со испуштање на дел од информациите од оригиналите (resampling) [TG96, Jel96, CS96, Tum96].

Комбинирањето на резултатите или редоследот и начинот на евалуацијата на поединте класификатори може да биде изведено на повеќе начини кои донекаде зависат и од природата на излезите на поединте класификатори и потребниот излез кој треба да се добие од комбинацијата. Според природата на излезната информација класификаторите може генерално да се поделат во три групи:

1. класификатори чиј излез претставува единствена класа во која е сврстен примерокот;
2. класификатори чиј излез претставува секвенца на сите можни лабели во кои може да биде сврстен примерокот, подредена според веројатноста на припадност на примерокот во соодветната класа, но не секогаш и квантитативна мерка на оваа веројатност;
3. класификатори кои на секоја од класите во која може да биде сврстен примерокот доделуваат квантитативна мерка која го изразува верувањето дека примерокот е припадник на токму таа класа.

Соодветно, зависно од видот на употребените класификатори постојат различни методи за нивно комбинирање. Наједноставната метода за комбинирање на резултатите од повеќе класификатори е по методот на гласање. Излезот од секој од класификаторите се брои како глас за одлука дека примерокот припаѓа на соодветната класа. Конечната одлука се носи во зависност од бројот на гласови кои ги собрала секоја од класите. Бројот на потребни гласови кои треба да ги собере класата за да се прогласи за победник како и условите кои треба да бидат задоволени околу распределбата на гласовите за воопшто да се прогласи победник (да не се отфрли примерокот како непрепознат) зависи од имплементираниот гласачки систем кој може да биде: гласање со консензус, гласање со просто или некое друго мнозинство [XKS92]. Кај класификаторите од третата група комбинирањето може да се врши по пат на усреднување, избор на најдобриот (winner-takes-all) или статистички подредено (order statistics) комбинирање на излезите од сите класификатори пред нивното проследување кон максимум селекторот [TG95, Tum96]. Доколку се располага и

со *a posteriori* веројатностите можно е при комбинирањето да се примени и Баесовото правило. Располагањето со квантитативна мерка за веродостојност (level of belief) на одлуката која ја носи секој од класификаторите, овозможува примена на т.н. тежинско гласање, односно вклучување на информацијата за тоа колку на кој од класификаторите може да му се верува при изведувањето на конечната одлука. Познавањето на ратите на препознавање, отфрлање и грешка на секој од класификаторите, овозможува примена на Dempster-Shafer критериумот при комбинирањето [XKS92]. Дополнителното познавање на конфузионите матрици (confusion matrix) за секој од класификаторите, отвара нови можности при комбинирањето. Располагањето со мерка на веродостојност која не зависи само од класификаторот туку и од класата за која тој се одлучил, односно познавањето за кои класи на примероци и во која мерка греши соодветниот класификатор, може дополнително да го подобри квалитетот на одлуката која се добива при комбинирањето. Мерката на веродостојност, ратите на препознавање, отфрлање и грешка и конфузионата матрица за секој од класификаторите може да се добие во процесот на проценката на перформанси на секој од класификаторите со примена на тестирачкото множество.

Освен организирани како комитети или ансамбли, класификаторите можат да бидат организирани и како група од експерти (mixture-of-experts) од кои секој е специјализиран за одреден поддомен од проблемот, а посебен класификатор наречен арбитер одлучува на кој од експертите ќе биде проследен проблемот [Bis95, KE96]. Stacked generalization методата пак опфаќа специфична организација на класификаторите во две нивоа, кои карактеристично се обучуваат [Bis95, FCS96]. Powalka *et al.* [PSW95a] разгледуваат неколку различни топологии на поврзување на повеќе класификатори кои ги делат на условни, хиерархиски и хибридни, а кои овозможуваат вклучување на дополнителни класификатори кога резултатот добиен од основниот класификатор ќе се процени како недоволно сигурен. Употребата на комбинација од класификатори може значително да го подобри препознавањето во системите за оптичкото препознавање на знаци. Подобрувањето на перформансите добиено со комбинирање на класификатори кај оптичкото препознавање на знаци е истражувано од страна на повеќе истражувачи [CST+92, Gos95, PSW95].

4.6 Контекстно постпроцесирање

Ненадминатите можности на човекот како супериорна “машина” за читање не се должат само на неговата природна способност за препознавање на примероци која е подобра од било кој вештачки класификатор. Значително влијание во процесот на читењето има фактот што човекот **го разбира** она што го чита. Ова разбирање го води процесот на читањето, односно читателот е способен да ги употреби лексичките, семантичките и синтактичките ограничувања присутни во текстовите и секоја буква да ја набљудува како дел од контекстот на зборот, секој збор како дел од фраза, реченица итн.

Дополнителните информации присутни во контекстот можат да бидат од значителна помош и при автоматското препознавање на текстови. Методите за користење на контекстот за подобрување на резултатите од препознавањето

може да се поделат на top-down, bottom-up и хибридни методи [Elm96]. Контекстот над кој делуваат пак, може да биде на ниво на парови или *n*-торки од знаци, зборови, парови или *n*-торки од зборови, фрази, реченици, пасуси, итн. Користењето на контекстот за подобрување на резултатите при препознавањето на текстови може да биде имплементирано во самиот процес на препознавање или како постпроцесорска детекција и корекција на грешки, избор од повеќе понудени алтернативи и пополнување на празнини. Контекстот може да се искористи за генерирање на хипотези кои го водат процесот на препознавање кој врши проверка на овие хипотези, како и за проверка на валидноста и избор на најсоодветната од алтернативите понудени од класификаторот. Контекстните знаења можат да бидат изразени во вид на статистички или синтактички правила.

Bottom-up методите претставуваат податочно водени пристапи кои јазикот го моделираат најчесто како Маркова верига. Веројатностите на транзициите меѓу состојбите, изведени со статистичка анализа на *n*-грамите во примерок од јазикот, можат да се искористат при донесувањето на конечната одлука при препознавањето со користење на Баесовиот критериум. Top-down методите се концепциски водени и имплементираат проверка на контекстната издржаност на резултатите добиени при препознавањето (проверка на валидни *n*-грами или зборови).

Експлоатирањето на контекстот најчесто се прави на ниво на збор (word-level processing). Располагањето со лексикон на сите можни зборови кои може да се појават во текстот овозможува проверка на валидноста на секој збор добиен како резултат од препознавањето. Доколку зборот не биде пронајден во лексиконот, можно е пребарување на лексиконот за пронаоѓање на алтернативи и избор на најсоодветната како замена за погрешниот збор. Рангирањето на алтернативите најчесто се врши според Levenshtein метриката, која растојанието меѓу два збора го изразува како минимална цена на операции на вметнување, бришење и замена на букви потребни за да се трансформира една низа од знаци во друга. Кај класификаторите кои нудат повеќе рангирани алтернативи за секој знак, можно е генерирање на повеќе кандидати на зборови од кои се избира валидниот, односно оној кои постои во лексиконот.

Детекцијата и исправањето на грешката при препознавањето не е воопшто едноставен проблем. За детекција на зборовите кои евентуално имаат грешки може да се искористат две информации: ниската надежност при препознавањето и/или непронаоѓањето на зборот во лексиконот кои сигнализираат веројатно погрешно препознат збор. Сепак овие две информации понекогаш не се доволно сигурни за оценување дека даден збор е погрешно препознат. На пример некој збор може да биде погрешно препознат и ако има релативно висока надежност или да биде препознат како некој друг збор кој постои во лексиконот. Колку и да е комплетен лексиконот тој не може да ги содржи сите можни зборови. Личните имиња, странските зборови, кратенките и акронимите иако можеби точно препознати би биле означени како сомнителни поради нивното отсуство од лексиконот. Дури и ако грешката при препознавањето е пронајдена, нејзиното поправање не е секогаш изводливо.

Генерално, постојат три вида на грешки при препознавањето на зборови:

1. препознавањето е погрешно, но точниот резултат е меѓу понудените кандидати;

2. препознавањето е погрешно, а нема понудено други кандидати;
3. препознавањето е погрешно, и точниот резултат не е меѓу понудените кандидати.

Грешките од првиот тип може да се поправат доколку се најде начин да се пронајде вистинскиот кандидат меѓу понудените. Останатите два вида грешки е многу потешко да се поправат. Доколку од листата на кандидати само еден е валиден, одлуката се носи во негова корист. Ако меѓу понудените кандидати повеќе се валидни, одлучувањето за најверојатниот (оној кој најчесто се појавува во јазикот) води кон правење најмала можна грешка. Пронаоѓањето на вистинскиот од листата на понудени кандидати не е воопшто едноставна работа особено во случаите кога два или повеќе од кандидатите се подеднакво веројатни. Во вакви случаи, за правилна одлука мора да се користи и контекстот на повисоко ниво од изолиран збор. Одлуката кој од кандидатите да се избере се носи во полза на зборот кој е најсоодветен во контекстот на соседните зборови или реченицата, врз база на лингвистички знаења [Hon95]. За испитувањето на контекстните зависности над нивото на изолиран збор постојат статистички и структурни пристапи кои веќе навлегуваат во областа на процесирање на природни јазици (NLP).

Статистичките методи вршат статистичко моделирање на јазикот преку експлоатација на статистички информации собрани од голем примерок на текст на јазикот кој се обработува. Типична и најчесто користена информација која директно се екстрахира е веројатноста за појава на два збора близу еден до друг (word collocation), која може да се користи при изборот на збор од листата на понудени алтернативи. Структурните пристапи се обидуваат синтаксата на јазикот да ја опишат со примена на лексикон, граматика и соодветен парсер. Граматиката претставува конечна спецификација на потенцијално бесконечниот број на реченици. Парсерот за дадената граматика врши анализа на дадена реченица и доделува структурни описи на реченицата врз база на граматиката, доколку дадената реченица може да биде карактеризирана преку дадената граматика. Во литературата може да се сретнат различни граматики за моделирање на англискиот јазик меѓу кои: контекстно слободни граматики, трансформациони граматики, лексички функционални граматики, генерализирани фразно структурни граматики и др., како различни методи за парсирање како ATN парсер, генерализиран LR парсер, детерминистички парсери, chart парсери и др. Во последниве години направени се повеќе обиди за обединување на статистичкиот и структурниот пристап, особено во областите на препознавање на говор, автоматско разбирање на текстови и говорен јазик и автоматското машинско преведување.

Taghava *et al.* [TBB+93, TBC94] изложуваат систем за постпроцесорска корекција на грешки во текстови добиени при оптичко препознавање на знаци, со користење на лексикон и генерирање на кандидати за замена на погрешно препознатите зборови со користење на конфузиона матрица која се гради за време на самиот процес на корекција. Подобрувањето на перформансите при оптичкото препознавање на знаци со употреба на лингвистички знаења (word collocation) е обработувано од страната на повеќе истражувачи [RE94, REL94, REJ94].

5 Систем за препознавање печатени македонски текстови

Во рамките на системот за препознавање на печатен македонски текст е развиена програма која за слика со скениран печатен текст, врши препознавање на текстот и го дава во излезна текстуална датотека. Шематскиот приказ на програмата за препознавање на печатен текст е прикажан на Слика 5.1. Влез во системот претставува датотека со сликата на веќе скенираниот текст во TIFF (Tagged Image File Format) формат. Бидејќи во нашиот пристап се користи top-down пристап за сегментација, сликата на текстот мора да биде правилно ориентирана (редовите да бидат хоризонтални). Затоа, на почетокот се врши корекција на закосеноста на сликата со текст. Сите натамошни обработки (од сегментирањето на блокови со текст до екстрахирањето на бинарните матрици на изолираните знаци) се вршат на корегираната слика со отстранета закосеност кај која редовите се хоризонтални во одредени граници на толеранција.

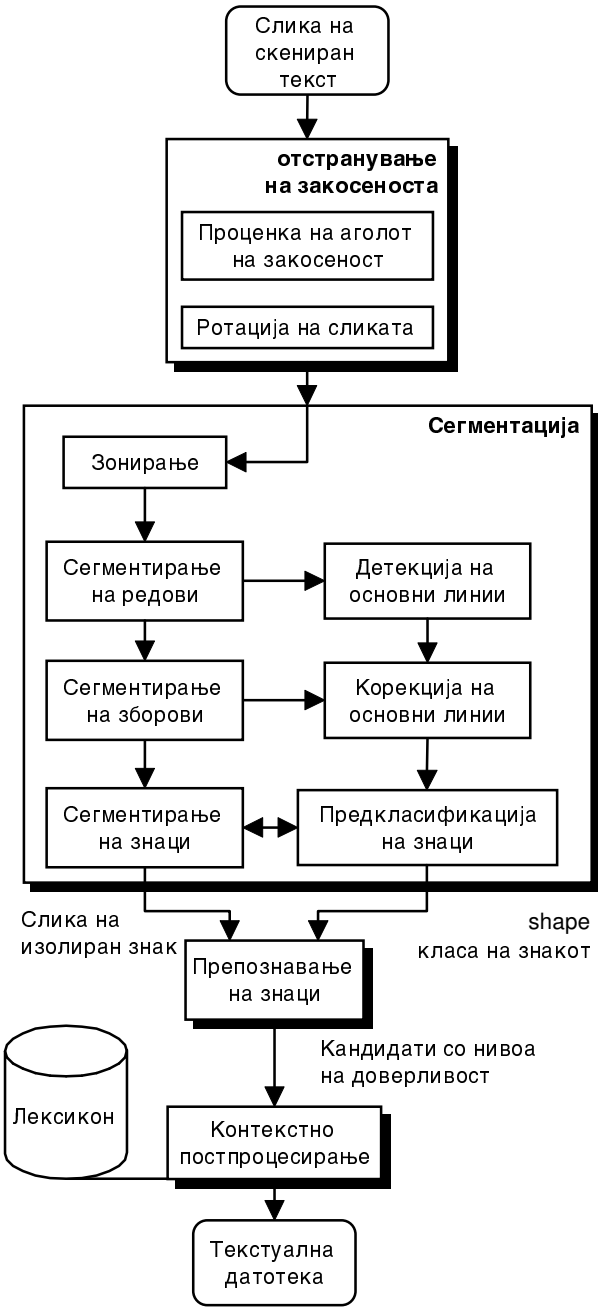
Сегментирањето се одвива во неколку фази кои се хиерархиски организирани:

1. Сегментирање на блокови со текст (зонирање) и одредување на редоследот на читање;
2. Сегментирање на редовите со текст во секој од блоковите;
3. Сегментирање на зборовите во секој ред со текст;
4. Сегментирање на знаците.

Во процесот на сегментација се врши класификација на знаците во одреден број на т.н. shape класи според нивниот облик – димензиите и поставеноста во редот (опишано во поглавјето 5.3). Како излез од процесот на сегментирањето се добиваат бинарни матрици со слики на изолираните знаци. Секоја слика на изолиран знак кон која е придружена и shape класата на знакот, се проследуваат на препознавање. Како класификатор се употребени адаптивни логички мрежи (Adaptive Logic Network – ALN) на чиј влез се носи самата бинарна матрица со сликата на знакот претходно скалирана во матрица со фиксни димензии зависни од shape класата на знакот. Изведени се повеќе ALN мрежи кои независно одлучуваат за припадноста на знакот. Резултатите добиени од сите ALN мрежи се комбинираат по пат на тежинско гласање по што се добива ранг листа на можните знаци (симболи) како кои може да биде препознае проследената слика, секој придружен со ниво на доверливост (веродостојност) (confidence level).

Листите со можни симболи (букви) за секоја знаковна позиција се проследуваат кон блокот за контекстно постпроцесирање во кој се врши нивно комбинирање и формирање на кандидат збор. Постоењето на зборот се проверува во лексикон, а неговото непостоење е сигнал за генерирање нов

кандидат со комбинирање на можните букви за секоја знаковна позиција во зборот. На крајот зборовите се запишуваат во излезна датотека која претставува симболичка претстава на сликата со текстот проследена на влезот.



Слика 5.1 Шематски приказ на системот за оптичко препознавање на печатен македонски текст

Системот за препознавање на печатен текст не комуницира директно со скенерот, туку сликата на скенираниот документ ја прима во вид на TIFF датотека од соодветната програма за скенирање која најчесто се испорачува со самиот хардвер за скенирање и е зависна од него. Фактот дека TIFF форматот е најчесто употребуван формат за сместување на скенираните податоци при работата со скенерите, придонесе кон одлуката тој да биде избран како основен формат во кој ќе се прима сликата на скенираниот текст, но системот може едноставно да се надогради и за примање на слики во други формати. TIFF

форматот за чување на слики е поддржан од скоро сите производители на скенери како и од најголемиот број на програми за графичка обработка на слики. Тој е супериорен во однос на повеќето други графички формати, но во споредба со нив и релативно покомплициран. TIFF [TIFF6] форматот претставува флексибилен и моќен формат за чување на дигитализирани слики, кој поддржува:

- Слики со големина до 65535 × 65535 пиксели;
- Сместување на црно-бели, слики во нијанси на сиво и слики во боја (во неколку модели на кодирање на боите);
- Сместување на повеќе слики во една датотека, како и на дополнителни неграфички специјалнонаменски или кориснички информации поврзани за сликата;
- Независност од платформата на која е создадена сликата;
- Компресија на сликите при што се поддржани повеќе методи за компресија;
- Воведување на класи кои претставуваат рестриктирани TIFF подмножества, кои ја поедноставуваат работата на корисникот.

TIFF формат на датотеки е базиран на т.н. тагови (ознаки), и дизајниран е за унапредување на размената на датотеки со дигитални слики. Тој е креиран како екстензибилен формат за размена на датотеки со слики, кој лесно би се надградувал. Примарната цел му е да овозможи богата околина за понатамошно усовршување на размената на датотеки на слики помеѓу апликативните програми. Најголем приоритет е даден на структуирањето на податоците, на начин кој овозможува минимизација на тешкотиите при идните дополнувања и усовршувања.

Моментално во системот за препознавање на печатен текст е изведена поддршка за класата B (Bilevel Images) TIFF датотеки. Бидејќи сликите кај ваквите TIFF датотеки се сместени како црно-бели слики, нема потреба од нивна бинаризација која е изведена при самото скенирање на текстот. Бинаризацијата која се изведува при самото скенирање зависи од софтверот за скенирање и можностите на хардверот за скенирање. Програмите за скенирање најчесто поддржуваат само униформна бинаризација, па во случаите кога е потребно препознавање на текст печатен на позадина со нерамномерна боја, ваквиот пристап не претставува задоволително решение. Сепак за најголемиот број случаи, работата со црно-бели слики на скенираниот текст нуди задоволителни перформанси, особено од аспект на заштедата во мемориски и процесорски ресурси кои се остваруваат при работата со нив.

5.1 Корекција на закосеност

Закосувањето на документот е типична деформација до која доаѓа при неговото скенирање. Тоа се должи на непрецизното поставување на документот на површината на скенерот, било тоа да се прави рачно или автоматски. Како резултат на оваа деформација, на сликата на скенираниот документ редовите со текст се поставени под одреден агол во однос на хоризонталата. Осетливоста на

применетиот алгоритам за сегментација на текстот на закосеност налага соодветно третирање на сликата пред нејзиното проследување на сегментација. Третманот се состои во ротација на сликата за соодветен агол, така да по оваа корекција редовите со текст се хоризонтално поставени (во одредени граници на толеранција кои не влијаат значително на перформансите на алгоритамот за сегментирање). Пред изведувањето на ротацијата неопходна е проценка на аголот на закосеност на редовите со текст.

5.1.1 Проценка на закосеноста на текстот

Доскоро комерцијалните програми за оптичко препознавање на знаци за персонална употреба воопшто не поседуваа можност за корекција на закосеноста, или доколку имаа таква можност, аголот за кој треба да се заротира сликата заради отстранување на закосеноста се задаваше од страна на корисникот. Дефинирањето на аголот на закосеност најчесто се изведува интерактивно со поставување на линија која се совпаѓа со правецот на простирање на редот со текст или со левата маргина над сликата на скенираниот текст. Последниве години, програмите за оптичко препознавање на знаци за персонална употреба присутни на пазарот стандардно овозможуваат автоматска детекција и корекција на закосеноста на скенираните текстови пред нивното препознавање.

Од аспект на целите поставени пред системот за препознавање на печатен македонски текст наменет за помош на лица со оштетен вид, кои треба да овозможи што поголема автономност (што помала интеракција со корисникот) автоматска детекција на закосеноста на скенираните текстови беше една од примарните задачи. Во нашиот систем е имплементирана автоматска проценка на закосеноста базирана на Hough трансформацијата која овозможува проценка на аголот на закосеност со точност од околу 0.25°.

Hough трансформацијата во својата генерализирана форма најчесто се употребува за детекција на прави, кружни или елипсовидни линии кај дигитализираните слики. Применета над слика со текст таа може да се употреби за детекција на доминантниот агол под кои се наклонети редовите со текст во однос на хоризонталата. Hough трансформацијата претставува линија-во-точка трансформација која овозможува лоцирање на прави линии на дигитализираните слики.

Права во декартов координатен систем може да се претстави со равенката:

$$y_i = ax_i + b \tag{5.1}$$

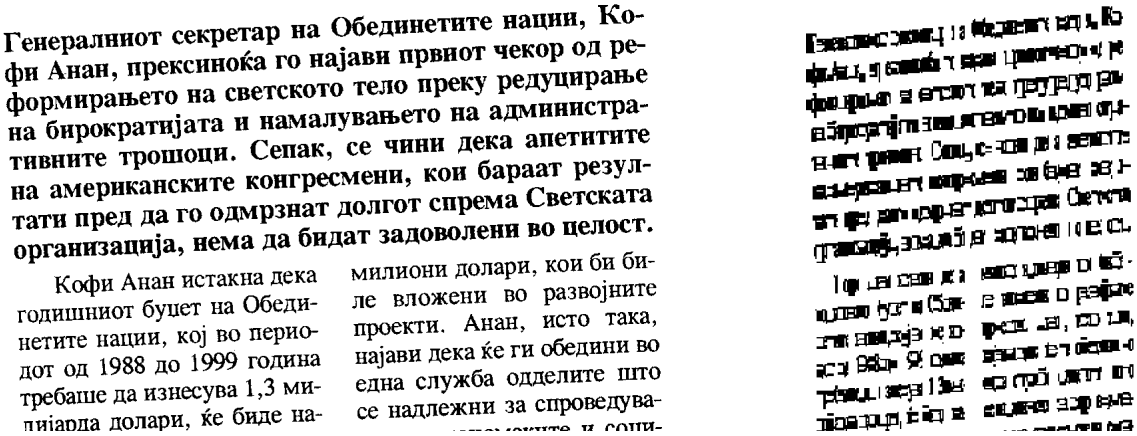
или во нормална форма, со равенката:

$$\rho = x_i \cos\theta + y_i \sin\theta \tag{5.2}$$

преку која за секоја права одредена со параметрите ρ – нормално растојание на правата од координатниот почеток и θ – аголот што го зафаќа нормалата од координатниот почеток кон правата со позитивната насока на x оската, се ограничуваат можните парови (x_i, y_i) кои ги одредуваат точките кои лежат на неа. Соодветно секоја линија од декартовиот координатен систем со равенката (5.2) се пресликува во точка (ρ, θ) во просторот на Hough трансформацијата. Слично,

секоја точка од декартовиот координатен систем се пресликува во крива (синусоида) во просторот на Hough трансформацијата, при што кривите кои произлегуваат од колинеарни точки се пресликуваат во криви кои се сечат во точката (ρ, θ), каде ρ и θ ги одредуваат параметрите на правата.

Во дискретниот случај, просторот на Hough трансформацијата претставува дводимензионално поле од акумулатори претставено со дискретни вредности на ρ и θ. За секоја црна точка на оригиналната слика се зголемуваат за еден соодветните акумулатори во трансформациониот домен. Локалните максимуми во точките (ρ_i, θ_i) во трансформациониот домен претставуваат линии одредени со параметрите (ρ_i, θ_i) на сликата. Вака имплементирана Hough трансформацијата претставува робусен метод за детекција на прави линии на дигиталните слики која е многу отпорна на шум, но е прилично захтевна во поглед на мемориските и процесорските ресурси.



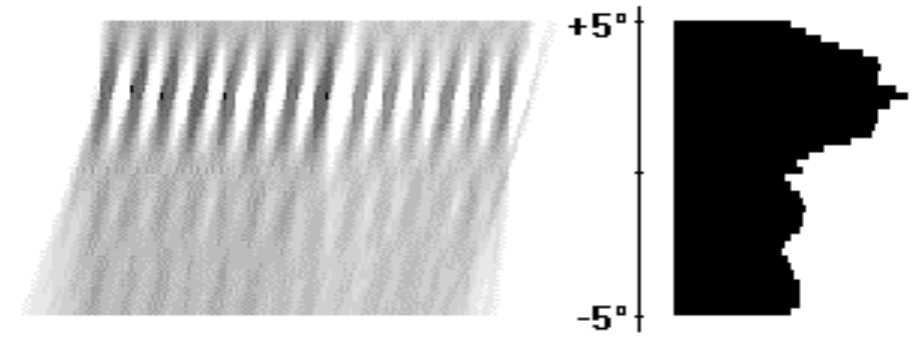
Генералниот секретар на Обединетите нации, Кофи Анан, прексиноќа го најави првиот чекор од реформирањето на светското тело преку редуцирање на бирократијата и намалувањето на административните трошоци. Сепак, се чини дека апетитите на американските конгресмени, кои бараат резултати пред да го одмрзнат долгот спрема Светската организација, нема да бидат задоволени во целост.

Кофи Анан истакна дека годишниот буџет на Обединетите нации, кој во периодот од 1988 до 1999 година требаше да изнесува 1,3 милијарда долари, ќе биде намилиони долари, кои би биле вложени во развојните проекти. Анан, исто така, најави дека ќе ги обедини во една служба одделите што се надлежни за спроведува-

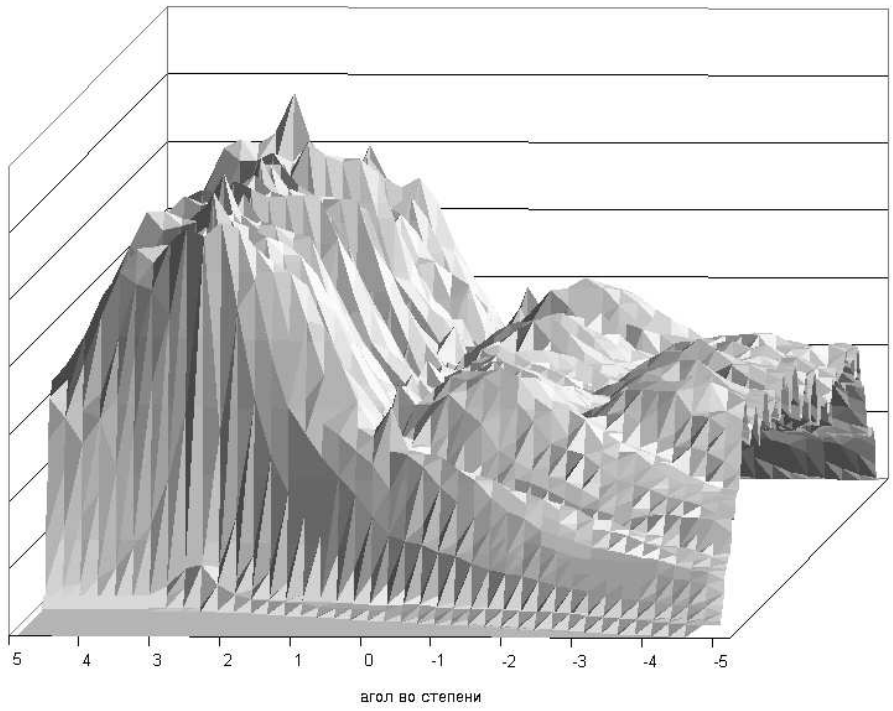
Слика 5.2 Оригинална и намалена слика на текстот за проценка на аголот на закосеноста

Затоа во нашата имплементација пред примената на Hough трансформацијата, оригиналната слика на документот прво се намалува, па потоа се пресликува во Hough доменот. Намалувањето зависи од димензиите на оригиналната слика и не мора да биде еднакво по хоризонтала и вертикала. Степенот на намалување се одредува зависно од димензиите на оригиналната слика со цел таа да се сведе на димензии помеѓу 265 од 512 точки по хоризонтала и 512 до 1024 точки по вертикала, што се покажа како доволно прецизно за одредување на аголот на закосеност. При тоа за степен на намалување по хоризонтала се избира најблиската од дискретните вредности: 2, 3, 4, 6 и 8, додека по вертикала степенот на намалување може да биде произволен цел број. Степенот на намалување по хоризонтала секогаш се избира да биде поголем или еднаков на степенот на намалување по вертикала. Со поголемото намалување по хоризонтала доаѓа и до зголемување на аголот на закосеност на текстот, а со тоа и на негова попрецизна проценка (Слика 5.2). При намалувањето доколку било која од точките од правоаголникот од $m \times n$ точки кои се пресликуваат во една точка во намалената слика е црна и резултантната точка во намалената слика ќе биде црна. Иако по ваквата трансформација намалената слика најчесто е нечитлива, таа е доволно прецизна и ги има сочувано основните особини на групираност на црните точки по должината на редовите со текст што овозможува одредување на аголот под кој е наклонет текстот.

Намалената слика со примена на Hough трансформацијата се пресликува во Hough доменот, при што θ оската се движи од –5° до +5°, а чекорот на дискретизација е 0.25°. Дискретизацијата по ρ оската е пропорционална на димензиите на намалената слика. При примената на Hough трансформацијата за проценка на аголот на закосеност на текст, редовите со текст практично ја имаат улогата на линии чија наклонетост се проценува, додека нивните нормални растојанија од координатниот почеток (како низпродукт на Hough трансформацијата) не се од интерес. По извршената трансформација, се бара глобалниот максимум во Hough доменот, чија θ вредност го претставува доминантниот агол на закосеност на редовите со текст (Слика 5.3).



(а) (б)



(в)

Слика 5.3 Проекција на сликата во Hough трансформационен домен и одредување на аголот на закосеност како глобален максимум во акумулаторското 2D поле: (а) Hough простор – потемна точка означува поголема вредност на акумулаторот; (б) Лоцирање на глобалниот максимум во Hough доменот; (в) 3D приказ на Hough доменот

Проценка на точноста на алгоритамот беше спроведена кај 78 слики со скениран текст закосен во границите од –5° до +5°. Грешката при проценката на

аголот беше помала од 0.12° кај 52 слики, во границите меѓу 0.12° и 0.25° кај 17, а кај сите останати слики освен кај една грешката беше помала од 0.5°. Интересно е да се забележи дека аголот беше нешто попрецизно одредуван кај сликите со едноколонски, отколку кај сликите со повеќеколонски текст. Од друга страна пак, сегментацијата (особено на редовите) кај повеќеколонски текстови е помалку осетлива на закосеноста отколку кај едноколонските текстови кај кои редовите се вообичаено подолги.

5.1.2 Корекција на закосеност

Корекцијата на закосеноста се сведува на ротирање на оригиналната слика околу нејзиниот центар за аголот на проценетата закосеност, само во спротивна насока. Иако на прв поглед тривијално, ротирањето е процесорски доста захтевна операција. Ротацијата претставува афина трансформација која врши еден спрема еден пресликување од слика во слика. Новите координати на секоја точка од изворната слика се добиваат со множење со матрицата на ротација (5.3).

$$\begin{bmatrix} x_r & y_r \end{bmatrix} = \begin{bmatrix} x & y \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \tag{5.3}$$

Ваквиот начин на пресметување вклучува по четири множења и две собирања за секоја точка. Со примената на т.н. scanline алгоритми може значително да се поедностават и забрзаат голем дел од геометриските трансформации меѓу кои и ротацијата. Scanline алгоритмите претставуваат специјална класа на геометриски трансформациони техники кои оперираат само по должината на редиците и колоните на сликата [Wol90]. Главната предност на овие алгоритми е нивната едноставност. Оперирањето над еднодимензионална низа од податоци ги поедноставува операциите (пресметките) и особено пристапот до меморијата. Така на пример shear трансформацијата по x оската изразена како множење со трансформационата матрица (5.4), со примена на scanline алгоритмите се сведува на поместување (шифтирање) на секоја хоризонтална линија со пиксели за одредена вредност. Фактот што сликите во меморијата најчесто се сместени во низи од последователни локации и што операцијата на поместување е поедноставна (и соодветно побрза) од аритметичките операции (особено множењето) доведува до значителни забрзувања при изведувањето на трансформациите имплементирани како scanline алгоритми. Scanline алгоритмите исто така се многу попогодни за хардверска имплементација.

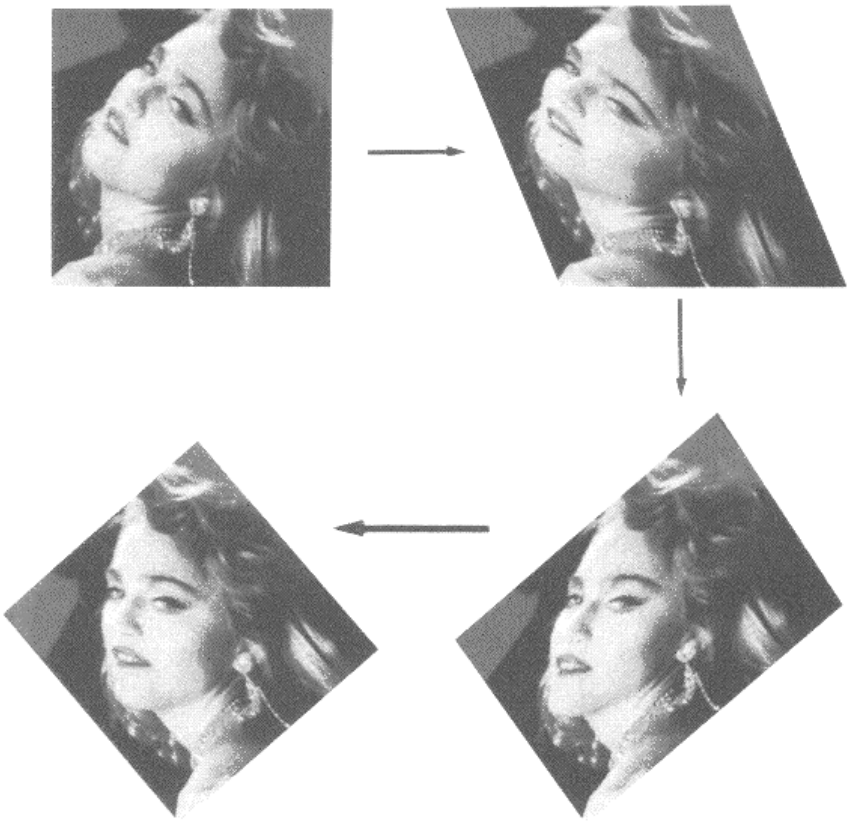
$$\begin{bmatrix} x_s & y_s \end{bmatrix} = \begin{bmatrix} x & y \end{bmatrix} \begin{bmatrix} 1 & 0 \\ S_x & 1 \end{bmatrix} \tag{5.4}$$

За да може дадена трансформација да се изведе со примена на scanline алгоритмите потребно е таа да биде сепарабилна, односно секоја димензија да може да се трансформира независно од другите. Ова значи дека трансформацијата треба соодветно да се разложи на неколку поедноставни трансформации од кои секоја оперира над само една димензија (или работи само по редици или само по колони). Предложени се повеќе различни разложувања на ротационата трансформација на поедноставни трансформации кои може

поефикасно да се пресметаат [BM80, CS80, Wei80]. Сепак најзначајна е методата до која независно дошле Paeth [Pae86] и Tnaka *et al.* [TK88] и која ротацијата ја разложува на три shear трансформации (5.5).

$$R = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -\tan\frac{\theta}{2} & 1 \end{bmatrix} \begin{bmatrix} 1 & \sin\theta \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -\tan\frac{\theta}{2} & 1 \end{bmatrix} \tag{5.5}$$

Според овој алгоритам сликата прво се изобличува по хоризонтала со една shear трансформација, потоа над вака добиената слика се применува shear трансформација по вертикала, за на крајот со уште една shear трансформација по хоризонтала да се дојде до ротираната слика (Слика 5.4).



Слика 5.4 Ротација со помош на три shear трансформации

Во нашиот систем за оптичко препознавање на знаци, корекцијата на закосеноста е имплементирана според алгоритамот на Paeth и Tnaka, при што првите две shear трансформации се изведуваат во едно минување: секоја хоризонтална линија од сликата прво се поместува (шифтира) за соодветна вредност, а потоа делови од неа се поместуваат во друга хоризонтална линија (поместување по вертикала). На крајот секоја линија уште еднаш хоризонтално се поместува за соодветната вредност.

5.2 Сегментација

Задачата на сегментацијата е на сликата со скенираниот текст да ги лоцира и издвои сликите на изолираните знаци кои го сочинуваат текстот. Исто така, во процесот на сегментацијата се одредува редоследот на читањето, односно редоследот по кој препознатите знаци ќе бидат сместени во излезната датотека. За таа цел прво се врши лоцирање на логичките целини на текстот (наслов, воведник, колони и пасуси со текст) што е познато како зонирање или сегментација на блокови со текст. За секој блок со текст се врши сегментација на редови со текст, потоа секој ред се дели на зборови и на крајот се изолираат поодделните букви кои го сочинуваат зборот. Проблемот на одредување на редоследот на читањето е практично присутен само при сегментирањето на блокови, бидејќи на ниво на блок со текст тој е одреден со правилата на употребеното писмо кои за кириличното како и за латинското писмо е по редови од горе надолу и во секој ред од лево на десно.

5.2.1 Сегментација на блокови со текст

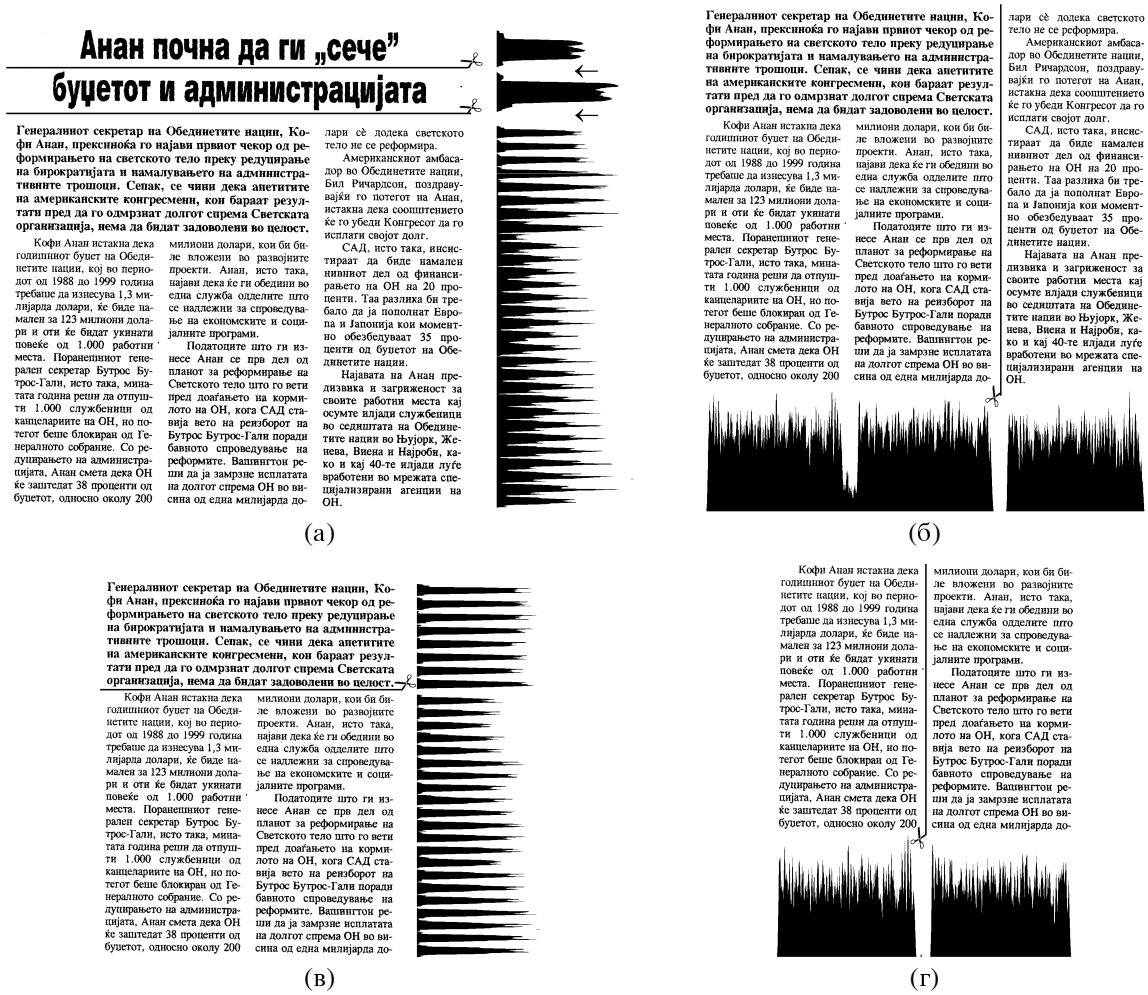
Под блок со текст се подразбира правоаголна област од сликата која содржи текст организиран во една колона, т.е. во кој редовите се простираат од левиот до десниот крај на блокот. Задачата на зонирањето е да ја разложи сликата на текстот на множество блокови со текст. Сегментирањето на блокови е изведено со адаптација на RXYC алгоритамот. Според овој алгоритам секој блок рекурзивно се дели наизменично по вертикала и по хоризонтала, изведувајќи ги поделбите на местата каде има хоризонтална односно вертикална белина. Лоцирањето на белините се изведува со анализа на хоризонталната, односно вертикалната проекција на блокот. Секоја длабока долина во низата на проекции поширока од одредена вредност е потенцијална локација за поделба на блокот. Поделбата почнува од почетен блок кој ја содржи целата слика со скенираниот текст. Алгоритамот прво се обидува да изврши поделба по вертикала (на неколку блокови кои се наоѓаат еден над друг). Ако не успее во ова тогаш сегментирањето ќе почне со обид за поделба по хоризонтала.

При поделбата по вертикала еден блок се дели на неколку блокови кои се поставени еден над друг (Слика 5.5 а). Секој од овие блокови, одејќи одгоре надолу, потоа се упатува на делење по хоризонтала (Слика 5.5 б). Блоковите што ќе се добијат со делењето по хоризонтала пак, повторно се упатуваат на делење но сега по вертикала одејќи од лево на десно (Слика 5.5 в). Секој новодобиен блок рекурзивно се упатува на ново делење по вертикала или хоризонтала зависно од тоа дали самиот бил добиен со делење по хоризонтала или вертикала соодветно. Ова се повторува сé додека блокот може да се дели (Слика 5.5 г). Доколку се оцени дека нема услови за понатамошно делење на блокот, тој се регистрира како таков и рекурзијата завршува. Условите кои треба да ги задоволи одредена правоаголна зона од сликата со текстот за да биде регистрирана како блок со текст се:

- да не е потесна од одредена минимална вредност;
- да не е пониска од одредена минимална вредност;

- да не е празна (процентот на пополнетост дефиниран како односот на црни точки и вкупниот број на точки во зоната да не е помал од одредена вредност).

Пред да се отфрлат зоните кои имаат мали димензии или пополнетост, се проверува нивното растојание до најблиската регуларна (регистрирана) зона и доколку се доволно блиску се обединуваат со неа, но само доколку со тоа не се наруши регуларноста на зоната кон која се припојуваат. Со ова се оневозможува отфрлање на блоковите кои содржат делови од некои букви кои се составени од одвоени компоненти (точката на j, цртичките на ќ и ѓ) особено кај крупните наслови кај кои може да се случи тие да бидат регистрирани како посебен блок со мала пополнетост. Со ограничувањето на минималните дозволени димензии за регистрација на валиден блок со текст се постига отфрлање на најголем број на концентрираните дамки од мастило кои се наоѓаат надвор од текстот, додека со контрола на пополнетоста на блокот се отфрлаат и зоните со расеани дамки.



Слика 5.5 Пример за сегментирање на блокови со аплицираниот RXYC алгоритам

Клучно при зонирањето на текстот според RXYC алгоритамот е пронаоѓањето соодветна локација каде треба да се изврши "сечењето". Ова се прави со испитување на хоризонталната проекција на зоната доколку се бараат локации за поделба по вертикала (Слика 5.5 а), односно на вертикалната проекција на зоната доколку се бараат локации за поделба по хоризонтала (Слика 5.5 б), соодветно. Поделбата се врши на средината на секоја долина (низа

од последователни проекции со мали вредности, обично блиски до нулата) поширока од одредена вредност. Граничната вредност на ширината на долината која ќе се смета за локација за поделба е различна за вертикалното и хоризонталното делење и се одредува динамички во зависност од димензиите на блокот кој се обработува. Така на пример, при поделбата на релативно висок блок по вертикала, дозволени се релативно мали широчини на долината (за правилна поделба на колони со текст кои се доста блиску), додека при поделбата на низок блок се бара поширока долина во вертикалната проекција (со што се избегнува поделбата на зборовите од крупните наслови во различни блокови).

Слика 5.6 Пример за одредување на редоследот на читање на блоковите со текст

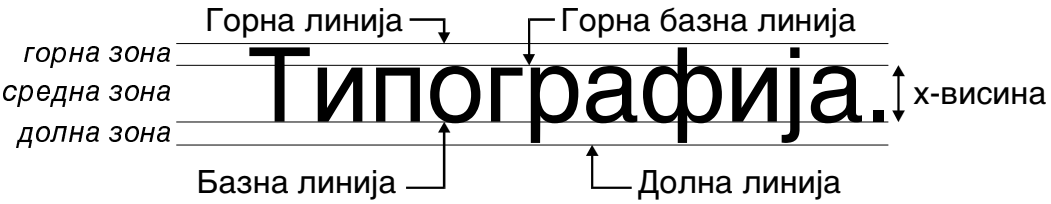
На крајот секој од блоковите со текст се редуцира на минималните димензии за да го опфати текстот, отфрлајќи ги белините по неговите граници. RXYC алгоритамот е згоден и заради тоа што ги дава блоковите во редоследот на нивното читање, односно редоследот на регистрација на блоковите всушност го претставува и редоследот на нивното читање (Слика 5.6).

5.2.2 Сегментација на редови со текст

Под ред со текст се подразбира блокот кој ја содржи сликата на еден ред со текст. Во еден блок со текст има еден или повеќе редови со текст кои се поставени еден над друг со меѓусебно растојание обично доволно за буквите од соседните редови да не се допираат. Карактеристично за редовите со текст е што тие се хоризонтални и нивната ширина (должина) е секогаш поголема од нивната висина. Во секој ред со текст има најмалку еден збор или дел од збор. Зборовите се наредени хоризонтално по должината на редот, со извесно меѓусебно

растојание, кое е поголемо од растојанието помеѓу одделните букви во рамките на еден збор. Зборовите се составени од еден или повеќе знаци (букви, цифри или специјални знаци) хоризонтално подредени по должината на зборот. Интересно е да се разгледа вертикалната поставеност на буквите во рамките на зборовите односно редовите.

Јасно е дека за да се разликуваат еден од друг различните симболи кои се користат во писмата мораат да имаат различни форми. Меѓутоа освен различните форми, во голем број писма знаците имаат и различни димензии и се пишуваат со специфична вертикална поставеност во однос на главните линии на хоризонталното простирање на редот со текст. Ова донекаде го отежнува одвојувањето на индивидуалните знаци, но дополнителните информации за вертикалната поставеност на знакот можат корисно да се употребат при неговото препознавање. Според вертикалното простирање, односно искористувањето на вертикалниот простор од хоризонтално ограниченият дел од редот што му припаѓа (му е доделен) на даден знак, знаците може да се поделат во неколку класи (обработено во поглавјето 5.3). Затоа, корисно е на некој начин да се одредат главните линии на простирањето на редот со текст во однос на кои би можела да се одреди вертикалната поставеност на знаците.



Слика 5.7 Дефинирање на карактеристичните зони за ред со текст

Четири хоризонтални линии ги дефинираат границите на трите карактеристични зони – горна, средна и долна, за секој ред со текст (Слика 5.7). *Долната линија* претставува физичка граница на редот со текст која го одвојува од редот под него. Соодветно *горната линија* е физичка граница на редот која го одвојува од редот над него. Овие линии се добиваат непосредно при самата сегментација на редовите, односно при одвојувањето на еден ред од редовите над и под него. Лоцирањето на *базната линија* и *горната базна линија* е клучно за дефинирање на зоните во рамките на редот со текст, во однос на кои се одредува вертикалното простирање на знаците. Кај најголемиот број пристапи кои го одредуваат и користат вертикалното простирање на знаците, лоцирањето на базните линии се врши дури по сегментирањето на знаците врз база на поставеноста и димензиите на рамките (bounding boxes) околу нив. Во нашиот пристап положбата на базните линии се прави при самата сегментација на редовите со користење на хоризонталната проекција, што овозможува оваа информација, поточно простирањето на знакот низ зоните на редот да се искористи при сегментацијата на знаци.

Сегментирањето на редови врши вертикално лоцирање на секој од редовите со текст во рамките на еден блок со текст. Сегментирањето на редовите е изведено со користење на хоризонталната проекција на блокот (Слика 5.8 б). Раздвојувањето на редовите настанува на местата на кои хоризонталната проекција има длабока долина. Редовите се манифестираат како изразени врвови

(peaks) во низата хоризонтални проекции, додека белините меѓу редовите како долини. Кај текстовите со оддалечени редови, овие долини се длабоки и лесно воочливи, но кај густо печатените текстови кои се малку закосени доаѓа до интерференција на проекциите од соседните редови, па лоцирањето на местата на кои треба да се изведе поделбата не е едноставно. Во нашиот пристап за сегментирање на редовите прво се бараат областите во кои хоризонталните проекции се под одредена вредност која се одредува во зависност од ширината на блокот и просечната големина на хоризонталните проекции. Потоа како локација за поделба се одредува минимумот во оваа област. При ова, може да се случи кај закосените густо печатени текстови проекциите на некои од редовите да интерферираат толку што минималната проекција меѓу нив не биде правилно детектирана и обата да бидат регистрирани како единствен ред (како на пр. седмиот и осмиот ред на (Слика 5.8 а). Исто така, може да се случи и еден ред да биде поделен на два (особено кратките редови кои содржат само еден збор со кои завршува пасус), или пак надредените знаци (точката на ј, и цртичките на ќ и ѓ) да бидат одвоени како посебен ред (особено кај крупните наслови).

(а) (б) (в)

Слика 5.8 Хоризонтална проекција на слика со текст и нејзината прва диференција

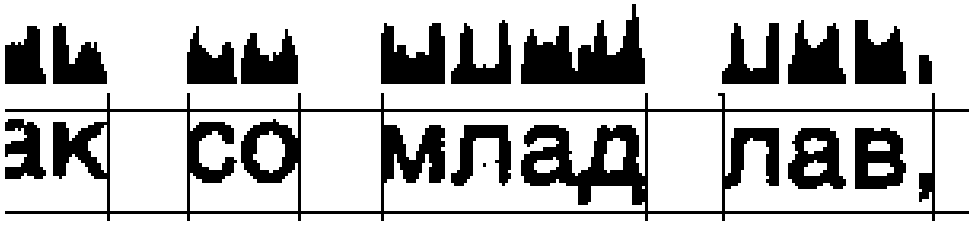
Кај најголем број документи, текстот во еден блок е напишан со иста големина на букви, па соодветно со голема веројатност може да се очекува висините на редовите во рамките на еден блок да бидат приближно константни (имаат мала девијација). Познавањето на просечната висина на редот, може да се искористи за исправање на грешките при сегментацијата на редовите со текст. Во нашиот пристап првата проценка на просечната висина на редовите со текст се прави уште пред нивната сегментација, со издвојување на доминантната фреквенција во низата на хоризонтални проекции. Врз база на оваа проценета вредност, по сегментацијата на редовите се врши нивна проверка и евентуална корекција. Редовите чија висина е над дозволената толеранција во однос на проценетата просечна висина на редовите, се упатуваат на повторна сегментација, но сега со полиберални ограничувања при барањето на локалниот минимум. Слично, за редовите со висина под одредена граница на толеранција се прави обид да се обединат со некој од соседните. По првата корекција се прави нова проценка на просечната висина на редовите со текст, при тоа не земајќи ги в предвид оние кои сеуште драстично отстапуваат од претходно проценетата

просечна висина. Потоа итеративно се врши повторна корекција и нова проценка на просечната висина на редовите сé додека има корекции на редовите.

По завршената корекција на редовите, за секој ред се одредува положбата на базната линија и горната базна линија. При лоцирањето на базните линии се користи хоризонталната проекција, поточно нејзината прва диференција. Алгоритамот за одредување на положбата на базните линии се потпира на претпоставката дека сите знаци (освен ' и ") се простираат низ средната зона, додека најчесто низ горната и долната зона се простираат релативно помал број на знаци, што ја прави средната зона најпополнета. Ова е изразено како нагол пораст и нагол пад на вредностите на проекциите, кој најдобро се детектира со анализа на првата диференција на проекциите (Слика 5.8 в). Во рамките на еден ред, максимумот во низата на први диференции одговара на локацијата на горната базна линија, додека минимумот (негативниот максимум) одговара на локацијата на базната линија. Претпоставката дека низ горната и долната зона се простираат релативно помал број на знаци во рамките на редот е точна за најголем број на редови, но отстапување има кај редовите во кои сите знаци се големи букви (наслови, нагласен текст). Во ваквите случаи одредувањето на горната базна линија не е можно бидејќи во редот нема ниту една мала буква која би ја одредила нејзината положба. Доколку е познато дека редот се состои само од големи букви, положбата на горната базна линија може да се претпостави како одреден процент од височината на големите букви, односно растојанието меѓу базната и горната линија. Совпаѓањето на горната линија со горната базна линија кое ќе се појави во вакви случаи може лесно да се детектира но тоа не значи дека редот е составен само од големи букви, бидејќи истото совпаѓање се појавува и кај редовите составени само од мали букви кои не навлегуваат во горната зона. Претпоставка дека редот е составен само од големи букви може да се изведе само на база на споредба на висината на редот со просечната очекувана висина на ред во блокот со текст, под претпоставка дека текстот во блокот е напишан со иста (типографска) големина на букви. Во нашиот пристап, кај ваквите редови ќе дојде до погрешна предкласификација на знаците, но овој проблем се третира во процесот на препознавањето и постпроцесирањето.

5.2.3 Сегментација на зборови

За секој ред со текст се врши негова сегментација на зборови. Сегментацијата на зборови подразбира хоризонтално лоцирање на почетокот и крајот на секој збор во рамките на редот со текст. Во нашиот пристап сегментацијата на зборови се врши со лоцирање на празнините во вертикалната проекција на редот (Слика 5.9) кои се подолги од одредена вредност која се одредува во зависност од x-височината на редот.



Слика 5.9 Сегментирање на зборови со користење на вертикалната проекција

Должината на празнината која ќе се смета за празнина меѓу зборови треба да се одреди така да е поголема од најголемото растојание меѓу буквите во рамките на еден збор, но не помала од најмалото растојанието меѓу два збора за да не дојде до двоење на зборовите и нивно регистрирање како два или повеќе одвоени зборови или обединување на два или повеќе зборови како еден. Погодна вредност за детекција на празнините меѓу зборовите може да се одреди со статистичка анализа на широчините на празнините во рамките на редот (Слика 5.10).

Слика 5.10 Распределба на хоризонталните растојанија меѓу знаците

Во редовите кои имаат мал број на знаци, статистичката проценка изведена само од информациите во рамките на редот не нуди квалитетни резултати. Во ваквите случаи за праг на ширина на празнина при сегментирање на зборовите се користи предефинираната вредност, која изнесува околу 50% од x-висината на редот, а која е добиена со статистичка анализа на поголеми количини на текстови.

5.2.4 Сегментација на знаци

Знаците се најмалите ентитети кои се лоцираат на сликата со текст. Растојанијата меѓу нив во рамките на зборовите се уште помали од самите димензии на знаците што дополнително ја отежнува нивната сегментација. Кај неизобличените печатени текстови знаците обично не се допираат и се поставени на одредено (мало) хоризонтално растојание еден од друг.

Сегментирањето на блоковите, редовите со текст и зборовите е релативно едноставно затоа што тие се релативно големи ентитети кои се одвоени со јасно видливи и релативно големи белини меѓу нив кои и при умерено деформирана и со шум оштетена слика на документот може да се лоцираат без поголем проблем сé додека закосеноста на сликата не е премногу голема. За разлика од ова, сегментацијата на знаците е доста осетлива на шум и на типичните изобличувања на печатените документи, додека не е толку осетлива на закосеноста на

документот. Како што се оди со хиерархиската сегментацијата според top-down пристапот на сé помали и помали ентитети така глобалните изобличувања на сликата (како закосеноста) стануваат помал проблем, а во прв план излегуваат проблемите со локалните изобличувања на сликата како шумот и типичните деформации кои настануваат кај печатените текстови (разлевање на мастилото, блед печат, и деформациите при умножувањето). Дури и најмалите деформации на документот можат да предизвикаат значителни изобличувања на знаците, како нивно допирање или распаѓање на повеќе поврзани компоненти што дополнително ја отежнува нивната правилна сегментација.

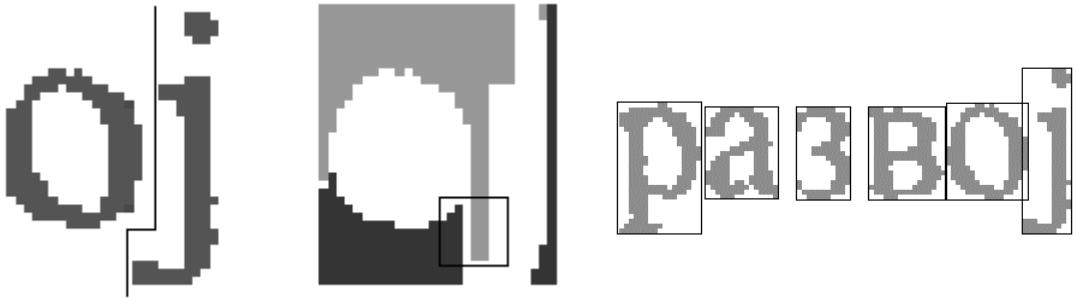
(а) (б) (в)

Слика 5.11 Пример за сегментирање на знаци – одгоре надолу: break cost функција, вертикална проекција, слика на зборот со основната (полни) и корегираната (испрекинати) проценка на базните линии, горниот и долниот профил

Сегментацијата на знаци во нашиот пристап се изведува по пат на дисекција со користење на вертикалната проекција, break cost функцијата и горниот и долниот профил како и информацијата за нивната припадност на класите на карактеристични форми (shape class). Локациите за дисекција на знаците се одредуваат преку барање на празнина во break cost функцијата и доколку празнината е широка точно еден пиксел, дисекцијата се врши на ова место, т.е. позицијата се одредува како крај на претходниот и почеток на следниот знак. Во случајот на поширока празнина, се проверуваат и вредностите на вертикалните проекции на тие локации и доколку и во нив постои празнина, крајот на претходниот знак се определува како позицијата на првата ненулева проекција лево од празнината, а почетокот на следниот како првата ненулева вертикална проекција десно од празнината (како на пример помеѓу е и в на Слика 5.11 б). Ако пак во вертикалната проекција нема празнина, дисекцијата се врши на локацијата на локалниот минимум во низата на вертикални проекции. На овој начин може да се сегментираат знаци кои не се допираат или малку се допираат и немаат значително преклопување.

Кај знаците кои се преклопуваат може да се случи проекцијата и break cost функцијата да немаат празнина иако знаците не се допрени (како меѓу о и ј на Слика 5.11 в). Во најголем број на случаи сегментирањето на ваквите парови може да се изведе со анализа на профилите. Имено, знаците кои се преклопуваат едниот од нив со некој свој дел навлегува во зоната на другиот (обично над или под него) не допирајќи го. Кај профилите ова се манифестира како минимална вредност на горниот профил помала од максималната вредност на долниот профил кој се наоѓа во неговото непосредно соседство (неколку колони пред или

по него) (Слика 5.12 б). Во ваквите случаи, поради преклопувањето на знаците одвојувањето на знаците не може да се изведе со обичен вертикален пресек. Кај ваквите парови, крајот на претходниот знак се наоѓа зад почетокот на следниот, па дисекцијата мора да се изведе со искршена линија (Слика 5.12 а). По сегментацијата на знаците за секој од нив се определува минималната рамка која го опфаќа знакот (Слика 5.12 в).



(а) (б) (в)

Слика 5.12 (а,б) Сегментирање на преклопени знаци со користење на профилите; (в) минимални рамки (bounding boxes) околу знаците

Проценката на базните линии која се прави при сегментацијата на редовите е точна само доколку редот со текст е хоризонтален. Кај долгите редови кои се и само малку закосени доаѓа до отстапувања кои се најизразени на почетокот и на крајот на редот. За отстранување на оваа непрецизност на проценката на базните линии, по сегментирањето на знаците се врши нивна корекција на ниво на изолирани зборови, за кои како ентитети со помала димензија од редот варијациите се помали. Ова се изведува користејќи ја првичната положба на базните линии и поставеноста на рамките на знаците во однос на нив. За таа цел се наоѓаат минималната и максималната вредност на положбата на горните и долните рабови на рамките на знаците од зборот, исклучувајќи ги при тоа од обработката специфичните знаци (' " , . – =) кои се детектираат според димензиите на рамките и нивното простирање низ зоните. Базната линија се поставува на вредноста на максимумот на долните рабови на рамките, а горната базна линија на минимумот на горните рабови. На Слика 5.11 а и б се прикажани сликите на два збора од кои првиот е од почетокот, а вториот од крајот на ред со текст закосен за –0.8°. Со полни хоризонтални линии е прикажана првичната положба на базните линии, а со испрекинати линии нивната корегирана положба.

По корекцијата на базните линии може да се спроведе предкласификација на знаците според формата на нивните рамки. По првичната сегментација на знаци можеби сé уште постојат неправилно сегментирани знаци, односно некои од допрените знаци да бидат регистрирани како еден знак или делови од распаднат знак да бидат регистрирани како два или повеќе одвоени знаци. Предкласификацијата на знаците според нивната форма е клучна за детектирање на сомнителните рамки и нивна повторна проверка со цел да се елиминираат допрените и распаднатите знаци.

Пронаоѓањето на сомнителните рамки се базира на проверка дали ширината на рамката е во опсегот на дозволени ширини за дадената класа, односно во зависност од зоните низ кои се простира рамката. За знаците кои се претесни се прави обид да се корегираат по пат на обединување со соседните.

Обединување се врши на два соседни тесни знаци кои се на релативно мало меѓусебно растојание, а при нивното обединување не би се добил преширок знак. Прешироките знаци се упатуваат на повторна сегментација користејќи ја вертикалната проекција. Прво се прави проценка на можниот број на знаци кои се допрени, а потоа поблиску се лоцираат опсезите во кои може да дојде до дисекција, земајќи ги во предвид можните ширини на допрените знаци. Дисекцијата се врши со пронаоѓање на минимумот во дадениот опсег на вертикалните проекции (разделување на допрените а и л на Слика 5.11). Одредувањето на границите на ширините за кои рамката се смета за валидна е доста тешко бидејќи постојат знаци (ш, ж, м) кои се значително пошироки од останатите, па понекогаш дури и пошироки од пар допрени знаци. Поместувањето на оваа граница надолу предизвикува детекција на поголем дел од допрените знаци, меѓутоа и евентуална непотребна сегментација на некој од пошироките знаци што ќе предизвика грешка. Со поместувањето пак на границата нагоре се избегнува непотребната сегментација на широките знаци, но поголем дел од допрените знаци можат да минат недетектирни. Затоа оваа граница треба да се одреди оптимално за да се минимизира грешката при сегментацијата. И покрај ова дел од знаците можат да минат погрешно сегментирани па затоа класификаторите се обучуваат при препознавањето да ги отфрлаат ваквите знаци, што може да се искористи како сигнал за нивна сегментација или обединување и повторно проследување на препознавање на корегираните знаци.

5.3 Предкласификација на знаците според формата

Задачата на предкласификацијата е да го разложи множеството на можни класи во кои може да се класификуваат знаците на одреден број помали по можност дисјунктни подмножества. Разложувањето на подкласи овозможува подобри перформанси, поедноставување на изведбата на класификаторите и лесно одредување дали се работи за голема или мала буква.

Нашиот систем е обучен да ги препознава следниве знаци:

- малите букви од а до ш;
- големите букви од А до Ш;
- цифрите од 0 до 9;
- специјалните акцентирани верзии на знаците é и ѝ;
- специјалните знаци ? % ( ) ! . , ' “ : ; + – / =

За разлика од латиницата кај која најголем број од знаците различно се пишуваат како голема и мала буква, кај кирилицата постојат само неколку букви кои имаат различна форма како мала и голема буква (Аа, Ее, Бб, Јј, Фф, Рр и Уу). При тоа, разликата во пишувањето на малата и големата буква е драстична само кај првите два пара (Аа и Ее), кај вторите два пара (Бб и Јј) постои извесна сличност, но разликите се доволни за буквите да можат да бидат препознати како различни знаци. Кај последните три пара (Фф, Рр и Уу) всушност се работи за многу слични облици, само што при пишувањето на малата буква таа навлегува и во долната зона. Сите останати букви (Вв, Гг, Дд, Ѓѓ, Жж, Зз, Ss, Ии, Кк, Лл, Љљ,

Мм, Нн, Њњ, Оо, Пп, Сс, Тт, Ќќ, Хх, Цц, Чч, Џџ, Шш) имаат скоро потполно иста форма напишани како мала и како голема буква. Според ова, од самиот облик на буквата набљудувана надвор од контекст и без информации за нејзините димензии и поставеност во однос на базните линии не е можно да се заклучи дали се работи за мала или голема буква. Од друга страна пак, повеќето парови мала-голема буква може да се третираат како единствен облик, односно да се класификуваат во единствена класа, а големината на буквата да се одреди според нејзините димензии и поставеност во однос на базните линии. Во нашиот пристап во процесот на предкласификација знаците се разврстуваат во 12 класи (означени како: **x**, **A**, **y**, **Q**, **j**, **!**, **.**, **,**, **'**, **–**, **:** и **;**) наречени shape класи. Разврстувањето во shape класите се изведува во зависност од формата на знаците, односно димензиите на нивните рамки и поставеноста во однос на базните линии [Spi95a, Spi95b, SS94, NS93].

Класа	Се простира низ зона			Знаци
	Долна	Средна	Горна	
A		✓	✓	АБВГҐЕЖЗЅИЈКЛЉМНЊОПРСТ ЌУФХЧШбќґéй 0123456789 ?!/%
x		✓		авгежзѕиклљмнњопстхчш :=-+.
y	✓	✓		д р у ц џ , ;
Q	✓	✓	✓	Д Ц Џ ф j ( )
‘			✓	' "

Слика 5.13 Класификација на знаците според нивното простирање низ зоните

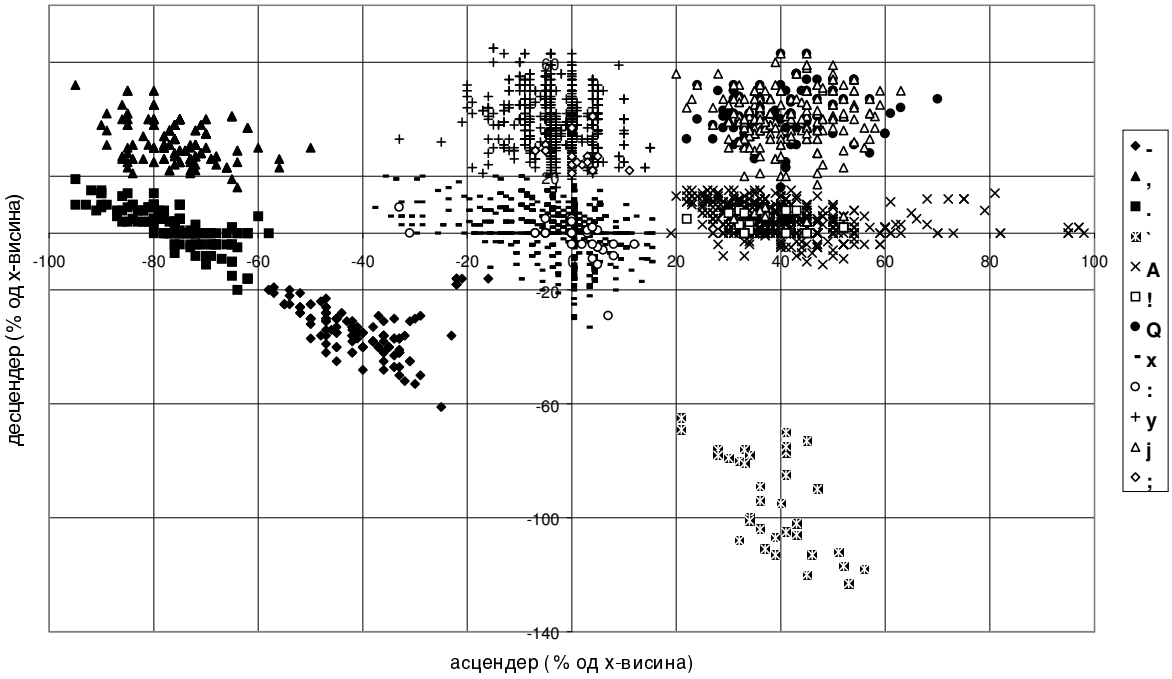
Класа	Знаци
A	АБВГҐЕЖЗЅИЈКЛЉ МНЊОПРСТЌУФХЧШ бќґ éй 023456789 %?
x	авгежзѕиклљ мнњопстхчш +
y	д р у ц џ
Q	Д Ц Џ ф
j	j ( )

Класа	Знаци
!	! 1 /
.	.
,	,
'	' "
–	– =
:	:
;	;

Слика 5.14 Класи на облици на знаците (shape classes)

Ако се изврши анализа на знаците од аспект на нивното простирање низ хоризонталните зони на редот со текст може да се изведе нивно групирање во 5 основни класи прикажани на Слика 5.13. Ако се земе во предвид и попрецизно определениот дел од зоните низ кои се простираат знаците како и нивните димензии, поточно односот на нивната ширина и висина може да се изврши групирање во споменатите 12 shape класи кои се прикажани на Слика 5.14. Бидејќи shape класите се базираат на формата на самите знаци, покажуваат голема независност од фонтовите.

Делот од знакот кој навлегува во долната зона е наречен и десцендер (descender), а делот кој навлегува во горната зона асцендер (ascender). Распределбата на асцендерот и десцендерот изразени како процент од x-висината на знакот, кај знаците од различни класи е прикажана на Слика 5.15. Негативните вредности на асцендерот и десцендерот означуваат дека знакот не само што не излегува од средната зона, туку и дека не ја пополнува во целост односно се простира само низ дел од неа (на пример знакот – го пополнува само средишниот дел од средната зона, додека знакот . само долниот дел). Од Слика 5.15 може да се види дека некои класи се јасно одвоени од другите класи, односно дека зафаќаат карактеристичен дел од асцендер-десцендер просторот, додека некои други класи се преклопуваат, односно не може јасно да се разграничат користејќи ги само асцендерот и десцендерот на знаците. Така на пример класите . , , – и ' се јасно разграничени од останатите класи и припадноста на знаците на некоја од овие класи може да се одреди само со познавање на асцендерот и десцендерот на знакот. Останатите класи се организирани во четири меѓусебно релативно јасно разграничени групи, од по две класи (x :, y ;, A ! и Q j) кои меѓусебно се преклопуваат. Кај знаците од овие класи, за да се одреди точната припадност кон одредена класа, неопходно е покрај асцендерот и десцендерот (кои можат да го лоцираат во една од споменатите групи) да се користи и информацијата за односот на ширината и висината на знакот.

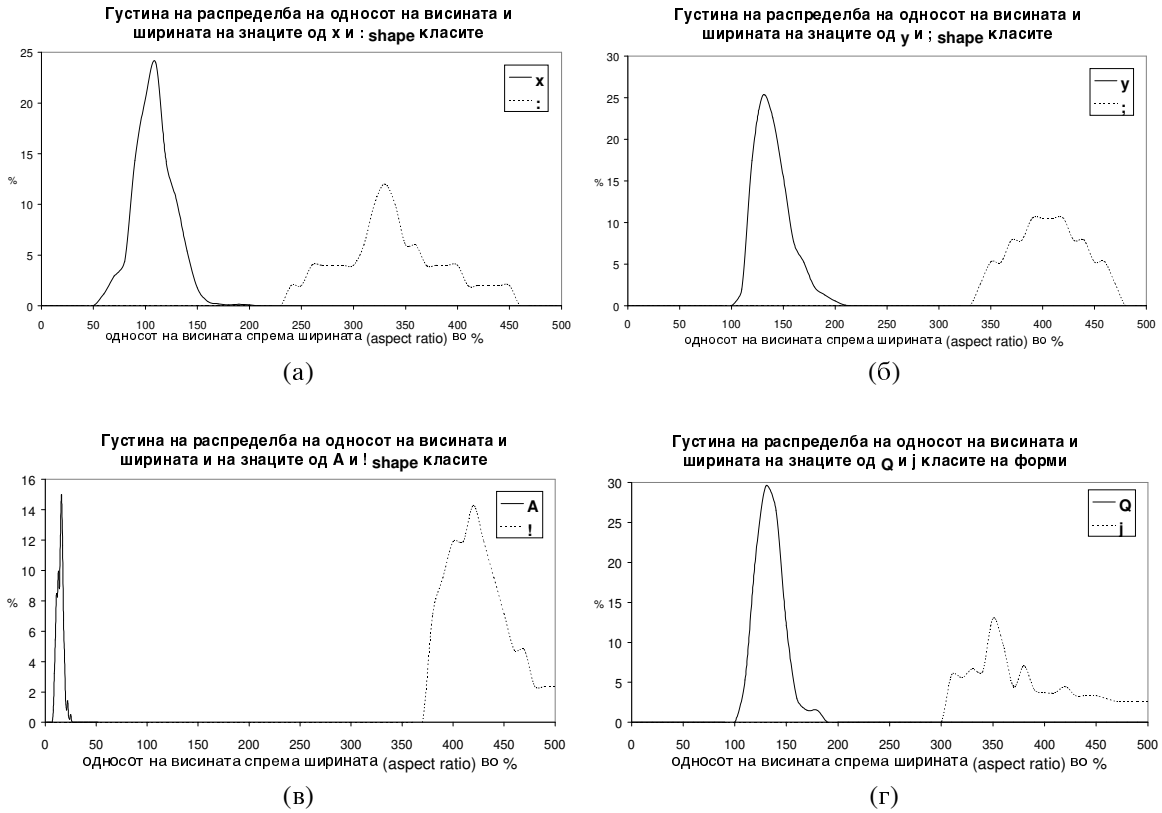


Слика 5.15 Распределба на класите знаци според асцендерот и десцендерот[1]

На Слика 5.16 а-г се прикажани густините на распределбите на односот на висината и ширината (aspect ratio) на знаците од паровите класи x :, y ;, A ! и Q j, од кои се гледа дека во сите четири случаи класите се меѓусебно јасно разграничени. Од ова може да се заклучи дека секој знак може едноставно да се распореди во една од дванаесетте shape класи, користејќи ги при тоа освен

[1] Распределбата е добиена со анализа на 7797 знаци изолирани од неколку скенирани текстови.

информациите за неговата вертикална поставеност во редот со текст (простирање низ зоните) и неговите димензии (односот на висината и ширината).



Слика 5.16 Густини на распределба на односите на висината во однос на ширината на знаците од паровите класи **x** **:**, **y** **;**, **A** **!** и **Q** **j**

Процесот на предкласификација во shape класи е имплементиран во вид на стебло на одлучување. Костурот на процесот на одлучување е даден на Слика 5.17. Горниот и долниот раб на рамката околу знакот се означени со **Ctop** и **Cbottom**, горната базна линија со **ubl**, базната линија со **bl**, ширината на знакот со **width**, а со **cc** е означена shape класата на знакот. Со **asc**, **desc** и **asp** се означени асцендерот, десцендерот и односот на ширината и висината на знакот кои се добиваат преку изразите (5.6), (5.7) и (5.8).

$$\mathbf{asc} = \frac{\mathbf{Ctop} - \mathbf{ubl}}{\mathbf{ubl} - \mathbf{bl}} \tag{5.6}$$

$$\mathbf{desc} = \frac{\mathbf{bl} - \mathbf{Cbottom}}{\mathbf{ubl} - \mathbf{bl}} \tag{5.7}$$

$$\mathbf{asp} = \frac{\mathbf{Ctop} - \mathbf{Cbottom}}{\mathbf{Cwidth}} \tag{5.8}$$

```
if(asc>0.15) {
   if(desc>0.15) {
     if(asp>3)              cc='j';
     else if(asp>0.7)       cc='Q';
   }
   else if(desc>-0.2) {
     if(asp>3)              cc='!';
     else if(asp>0.55)      cc='A';
   }
   else if(desc<-0.5)       cc='`';
 }
 else {
   if(asc<-0.6)
      if(desc>0.2)          cc=',';
      else                  cc='.';
 else if(desc>0.2)
   if(asp>300)              cc=';';
   else if(asp>100)         cc='y';
   else if(asc<-0.15 && desc<-0.15 && asp<0.8)
                            cc='-';
   else if(asp>2)           cc=':';
   else if(asp>0.5)         cc='x';
 }
```

Слика 5.17 Процес на одлучување при одредување на shape класата

Со предкласификацијата на знаците според обликот се постига намалување на димензионалноста на проблемот. Поделбата на знаците во неколку shape класи од кои секоја има помал број знаци, овозможува проблемот на препознавање на знаци да се подели на различни типови на подпроблеми, односно откако ќе се определи shape класата на знакот тој да се упати на препознавање на експерт специјализиран за препознавање на знаци од специфичната класа. Работата на ваквите експерти е поедноставена бидејќи постои помал број на можни исходи на препознавањето, па соодветно нивната конструкција, обучување и перформанси се подобри во однос на евентуалниот универзален препознавач кој би требал да ги препознава сите можни знаци. Поделбата на shape класи исто така овозможува конструкција на специфични класификатори оптимизирани за соодветната класа. Кај shape класите кои се составени од еден единствен знак (**.**, **,**, **:**, **;**), препознавањето е практично завршено со самата предкласификација. Сепак предкласификацијата токму за овие парови на знакови (., и :;) е најнепрецизна, па во нашиот пристап и знаците од овие класи се проследуваат на класично препознавање.

5.4 Препознавање на изолирани знаци

Задачата на препознавањето на изолирани знаци е за дадена слика на изолиран знак (бинарна матрица во нашиот случај) да одговори за кој знак станува збор, односно да го разврсти во една од симболичките класи од знаците кои ги препознава системот или евентуално да го отфрли како непрепознатлив. Во нашиот систем класификаторот е изведен со помош на адаптивна логичка мрежа, поточно повеќе адаптивни логички мрежи кои со тежинско гласање

конкурентно одлучуваат за припадноста на даден знак кон одредена симболичка класа.

5.4.1 Реализација на системот за препознавање печатени знаци

Системот за препознавање на печатени знаци треба секој проследен примерок (сликата на знакот) да го распореди во една од симболичките класи. Симболичките класи се изведени од знаците кои треба да бидат препознавани, но меѓу нив не постои еден спрема еден кореспонденција, односно една симболичка класа може да содржи и повеќе знаци. Така на пример големите и малите букви кои имаат иста форма се распоредени во единствена класа, што значи дека класификаторот ќе ја препознае само буквата но не и дали таа е голема или мала, што лесно може да се одреди од shape класата на знакот. Исто така некои од цифрите кои имаат исти или слични форми со некои букви се сместени во единствена симболичка класа, (на пример буквите о и О и цифрата 0, буквите З и з и цифрата 3, како и буквите б и цифрата 6). При распоредувањето на примерокот во некоја од овие симболички класи, доколку тој бил од shape класата **x** јасно е дека станува збор за малата буква, но доколку бил од shape класата **A** одлуката дали се работи за голема буква или цифра може да се донесе само по испитување на контекстот во кој се наоѓа. Знаците кои ги препознава системот се распоредени во 63 различни симболички класи, но по потреба лесно можат да се додадат и нови класи.

Подсистемот за препознавање печатени знаци е изведен со примена на адаптивни логички мрежи. Бидејќи адаптивните логички мрежи имаат само еден излез кој е бинарен, една мрежа може да изврши класификација на проследениот примерок во една од само двете можни класи. За проблемот на класификација на непознатиот знак во една од 62-те симболички класи очигледно е дека една ALN мрежа не може да ги задоволи барањата. Затоа во нашиот пристап се употребуваат најмалку онолку различни одвоени ALN мрежи колку што има и класи, при што секоја мрежа е специјализирана да ги препознава знаците од нејзината симболичка класа, односно да одговара на прашањето дали даден примерок припаѓа или не на нејзината класа. Непознатиот примерок се проследува на препознавање на повеќе мрежи, а на крајот само една би требала да го препознае. Изведбата на класификаторот вклучува и постоење на повеќе ALN мрежи за иста симболичка класа наречени *гласачи* кои кооперативно одлучуваат за припадноста на знакот кон одредена класа, при што секој од гласачите дава или не дава глас за припадност на непознатиот знак кон неговата класа. На крајот гласовите се собираат и победата ја носи онаа класа која собрала најмногу гласови, т.е. знакот се распоредува во таа класа.

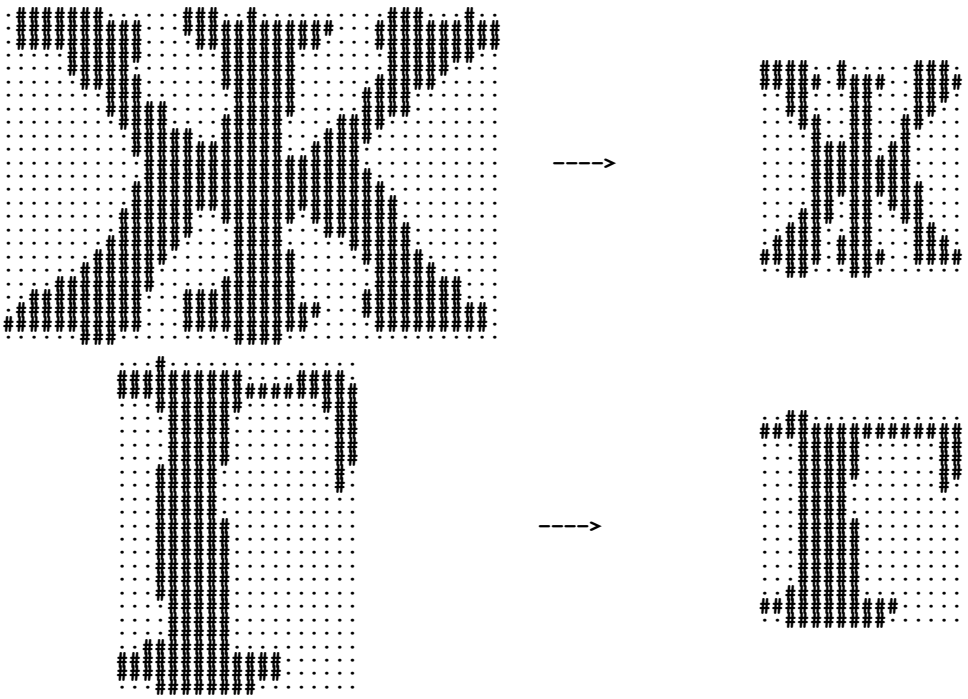
Скалирање на примерокот

За да може изолираниот знак да се проследи на препознавање на адаптивна логичка мрежа, потребно е да се претвори во бинарен вектор со фиксна димензија. За таа цел бинарните матрици со различни димензии кои зависат од големината на буквата прво се сведуваат на матрица со фиксна големина, а потоа од неа се прави вектор, со едноставно нанижување на редиците една по друга. Димензиите на кои се скалира секој од знаците зависат од shape класата на

знакот, а се избрани така за да го сочуваат просечниот однос на ширината и висината за знаците од дадената shape класа и се поставени на вредностите прикажани на Слика 5.18. Соодветно и ALN мрежите одговорни за препознавање на знаците од соодветните shape класи се конструирани да примаат вектор со големина која одговара на големината на соодветната целна матрица.

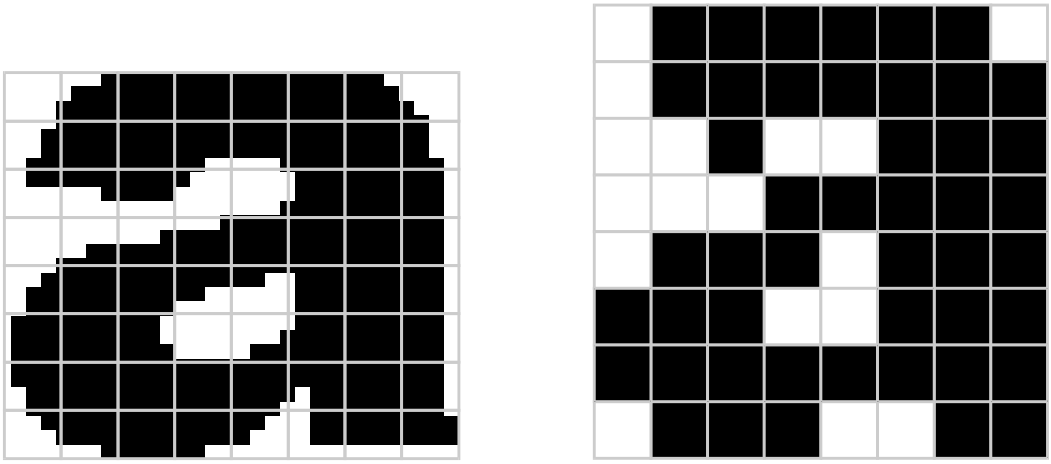
Класа	Висина	Ширина
A x –	16	16
Q y	24	16
j !	24	8
. , '	8	8
: ;	16	8

Слика 5.18 Димензии на целната матрица при скалирањето, зависно од shape класата



Слика 5.19 Знаците од иста shape класа се скалираат на исти димензии

Знаците може да се скалираат за различен фактор по ширина и по висина со цел максимално да се пополни габаритот на целната матрица односно да се елиминираат празнините кои би се појавиле лево и десно или горе и доле од знакот кога скалирањето би било со ист фактор и по вертикала и по хоризонтала. Со ова се постига пренесување на поголем дел од корисната информацијата за формата на знакот и истовремено се елиминираат варијациите во ширините и висините на знаците од иста shape класа (Слика 5.19). Отстапување од ова има само кај знаците од shape класата – кај која факторот на скалирање е ист по вертикала и хоризонтала, а се одредува така да се пополни целиот хоризонтален простор на целната матрица.



Слика 5.20 Скалирање на знак со препокривачка решетка

Скалирањето се спроведува со препокривање на оригиналната матрица на знакот со решетка со димензии исти со димензиите на целната матрица. Потоа за секој отвор во решетката се одмерува црната површина и доколку таа е поголема или еднаква на половината од целокупната површина на отворот, соодветната позиција во целната матрица се пополнува со 1, инаку со 0 (Слика 5.20). Зависно од димензиите на појдовната и целната матрица при скалирањето може да дојде и до зголемување и до намалување по едната или обете димензии на знакот.

Комбинирање на класификаторите

Комбинирањето на класификаторите го обработува проблемот како од одлуките на повеќе независни класификатори да се изведе единствена одлука, односно изведување на нов настан $E(x)=y$, за дадено множество на настани $e_к(x)=y_k$, $k=1,\ldots,K$ каде равенката (5.9) не мора да биде исполнета.

$$e_1(x) = e_2(x) = \cdots = e_k(x) \tag{5.9}$$

За проблемот на препознавање на примероци x го претставува непознатиот примерок кој треба да се препознае, $E(\cdot)$ функцијата на комбинирање која примерокот x го распоредува во симболичката класа y користејќи ги резултатите од K независни класификатори $e_к(\cdot)$ од кои секој примерокот x го распоредил во симболичката класа y_k соодветно.

ALN мрежите имаат само еден бинарен излез па соодветно секоја од нив произведува еден бинарен настан. Нека располагаме со N_v гласачки ALN мрежи за секоја од N_c класите кои треба да се распознаваат. Во тој случај за секој примерок проследен на препознавање на сите $N_v \cdot N_c$ ALN мрежи би добиле вектор од $N_v \cdot N_c$ бинарни вредности:

$$e_i^j(x) = \begin{cases} 1 \\ 0 \end{cases}, \quad i = 1,\ldots,N_v; \quad j = 1,\ldots,N_c \tag{5.10}$$

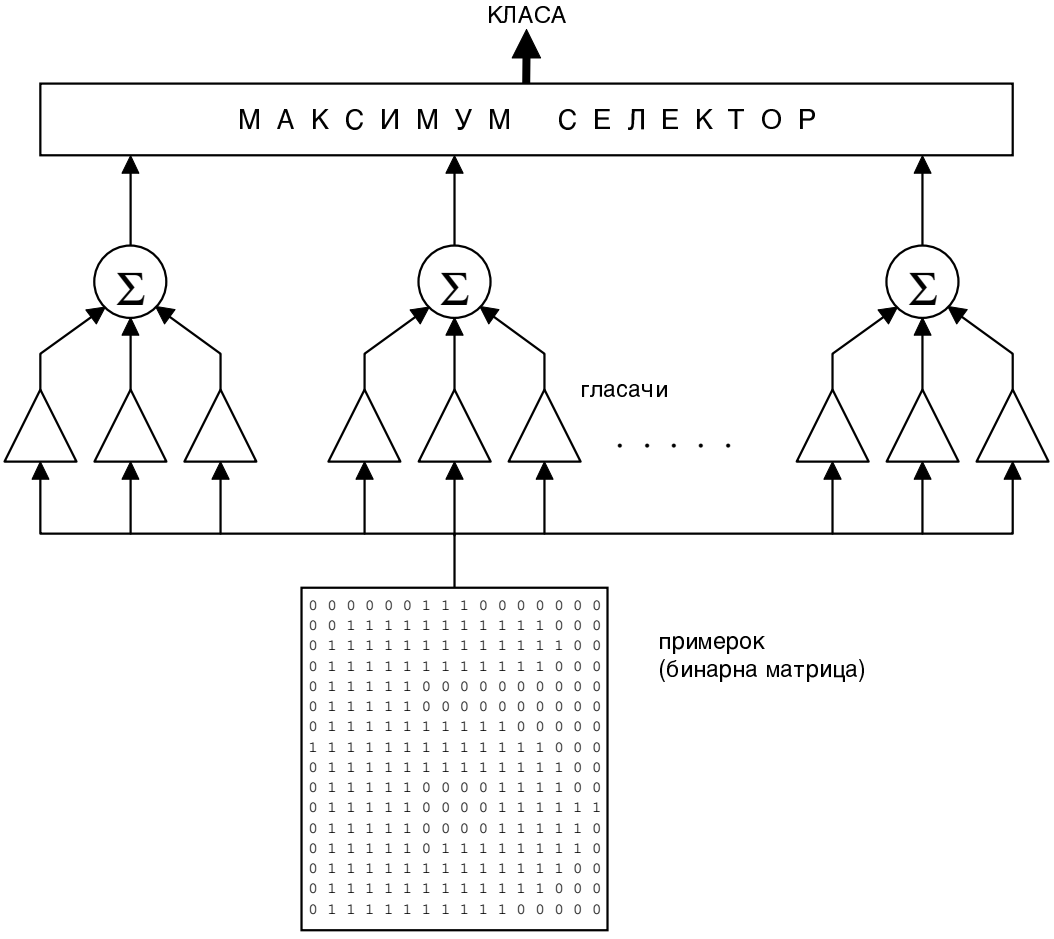
Одредувањето на настанот $E(x)$, односно на која класа да се додели примерокот x наједноставно се изведува по пат на гласање. Начинот на гласање може да биде различен. Така на пример, при гласање со консензус се бара примерокот да го препознале **сите** гласачи од **само една** класа и никој друг:

$$E(x)=\begin{cases} y, & \text{ако } \sum_{i=1}^{N_v} e_i^y(x)=N_v \underset{\substack{s=1\\ s\neq y}}{\overset{N_c}{\wedge}} \sum_{i=1}^{N_v} e_i^s(x)=0 \\ \phi, & \text{инаку} \end{cases} \tag{5.11}$$

каде ϕ е ознака дека примерокот се отфрла како непрепознат. При просто мнозинско гласање примерокот се класификува како y ако и само ако бил препознат од повеќе од половината гласачи кај само една класа:

$$E(x)=\begin{cases} y, & \text{ако } \sum_{i=1}^{N_v} e_i^y(x)>\frac{N_v}{2} \underset{\substack{s=1\\ s\neq y}}{\overset{N_c}{\wedge}} \sum_{i=1}^{N_v} e_i^s(x)<\frac{N_v}{2} \\ \phi, & \text{инаку} \end{cases} \tag{5.12}$$

Во нашиот систем при препознавањето примерокот се упатува на препознавање на мрежите кои се одговорни за препознавање на знаците од shape класата на која и припаѓа примерокот, при што за секоја симболичка класа постојат по неколку гласачи (Слика 5.21).



Слика 5.21 Препознавање на знаци со употреба на еднаков број на гласачи по симболичка класа

Во првата имплементација бројот на гласачи за секоја симболичка класа беше ист. Во ваков случај одлуката за припадност на примерокот кон одредена класа се носи со гласање по методот на просто мнозинство. Излезите од сите мрежи (гласачи) за одредена симболичка класа се собираат и се проследуваат кон

максимум селекторот кој ја одбира класата со најмногу гласови. Доколку бројот на гласови кои го собрала одредена симболичка класа од сите свои гласачи е поголем од половината од бројот на гласачи и истовремено кај сите останати класи бројот на гласови кои го собрале е помал од половината од бројот на гласачи, се одлучува во корист на симболичката класа која собрала најмногу гласови. Во случај ниту една од класите да не собрала доволно гласови примерокот се отфрла како непрепознат. Истото се случува и доколку две или повеќе класи соберат повеќе од пола од можниот број гласови. На сличен начин може да се изведе и двотретинско или некој друг вид на мнозинство, зависно од сигурноста со која сакаме да го препознаеме примерокот но и од процентот на отфрлени знаци кои сме подготвени да го прифатиме. Со зголемување на минималниот број на потребни гласови кои треба да бидат собрани од симболичката класа за да и се додели примерокот, се намалува процентот на погрешно препознати знаци, но се зголемува процентот на отфрлени знаци. Со зголемувањето на бројот на гласачи исто така се зголемува и точноста на препознавањето која е подобра од точноста на најдобриот од гласачите.

Во апликации во кои процентот на отфрлени знаци треба да е што помал може да се спроведе следново одлучување: секогаш се одлучува за класата која собрала најголем број гласови, независно од тоа колкав е тој број и каква е распределбата на гласовите кај останатите класи. При ваков систем на гласање за отфрлање ќе останат само случаите во кои ниту една мрежа не го препознала примерокот или две или повеќе класи собрале ист број гласови. Во ваков случај, бројот на отфрлени примероци ќе биде минимален, но и бројот на супституции (погрешно препознати знаци) ќе биде поголем.

Доколку располагаме со систем за контекстно постпроцесирање, кој е во состојба да детектира и евентуално да поправи дел од грешките при класификацијата на знаците, подобро е на овој систем да му се проследат сите кандидати од кои класификаторот не можел да се одлучи, па постпроцесорот на база на контекстот да одлучи за најверојатниот знак. Во ваков случај излез од класификаторот повеќе не треба да биде само една (или ниедна, ако примерокот не е препознат) симболичка класа, туку цела низа на можни класи на кои може да припаѓа знакот, секоја придружена со соодветна веројатност дека примерокот припаѓа баш на таа класа. Со ова практично се проследува скоро целата (делот зависно од бројот на кандидати кои се проследуваат) нефилтрирана информација од класификаторот и конечната одлука му се препушта на контекстниот постпроцесор.

Во претходниот случај комбинирањето на класификаторите ја даваше како резултат најверојатната симболичка класа на која припаѓа примерокот. Во ваков случај примерокот се доделува на точно една класа иако можеби имало гласови (но помалку) за негово распоредување и во други класи. За да се изрази неспецифицирана (fuzzy) припадност на примерокот кон повеќе од една класа, потребно е во зависност од бројот на гласовите да се одреди веројатноста дека примерокот е припадник на дадена класа K_c, дадена со (5.13) кој обезбедува веројатностите да сумираат на 1 (5.14).

$$P(x \in K_c) = \begin{cases} \dfrac{\sum_{i=1}^{N_v} e_i^c(x)}{\sum_{i=1}^{N_v}\sum_{s=1}^{N_c} e_i^s(x)}, & \text{ако} \quad \sum_{i=1}^{N_v}\sum_{s=1}^{N_c} e_i^s(x) \neq 0 \\ 0, & \text{ако} \quad \sum_{i=1}^{N_v}\sum_{s=1}^{N_c} e_i^s(x) = 0 \end{cases} \tag{5.13}$$

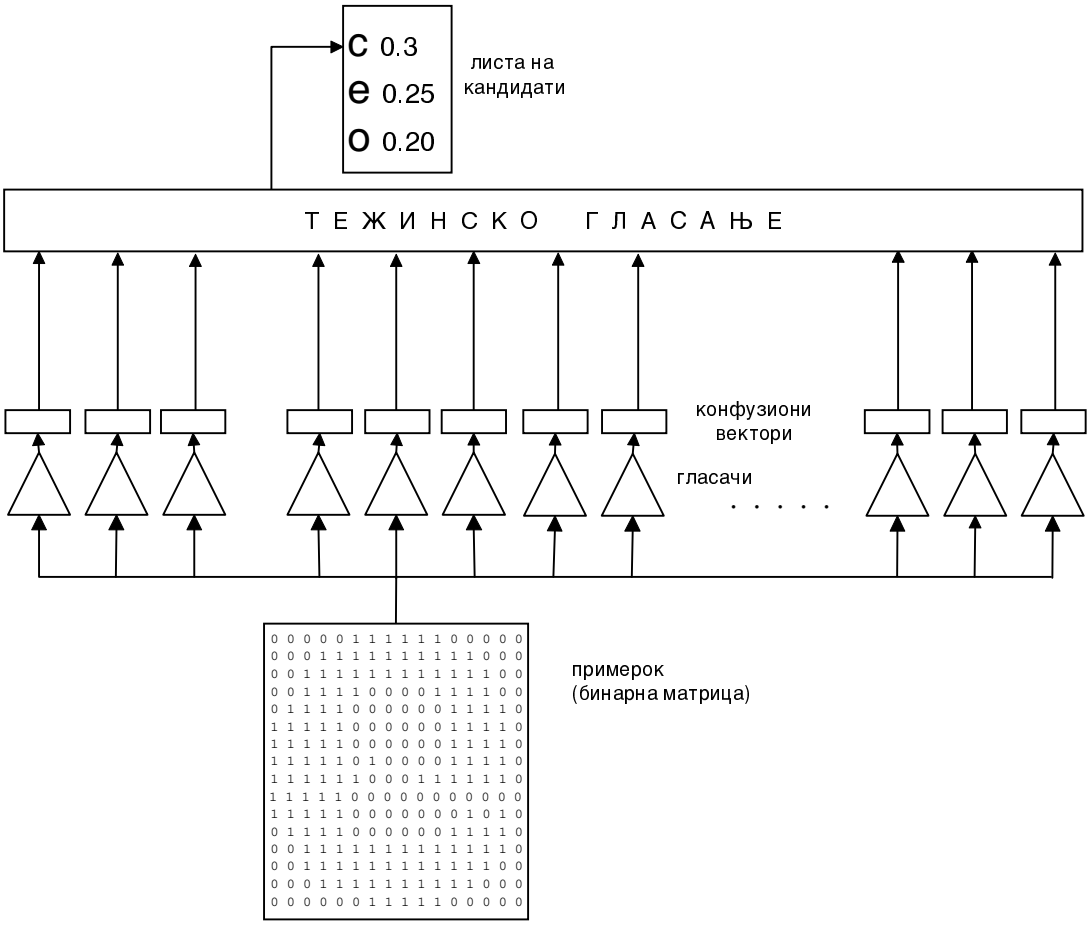
$$P(x \in \phi) + \sum_{s=1}^{N_c} P(x \in K_c) = 1 \tag{5.14}$$

Вака пресметаните веројатности ја изразуваат веројатноста примерокот да припаѓа на одредена класа земајќи го при тоа во предвид распоредот на гласовите по класите, но не и вкупниот број на гласови. Така на пример ако за секоја класа постојат по 5 гласачи и ако при класификацијата на даден примерок само две од нив добијат по 1 глас, веројатноста дека примерокот е припадник на секоја од тие две класи е 0.5. Истата веројатност ќе се добие и ако обете класи добиле по 5 гласа. Бидејќи за потребите на контекстниот постпроцесор е потребна некој вид на мерка која ќе ја изразува сложената веројатност дека дадениот примерок е припадник на дадена класа ако е воопшто правилно препознат, посоодветно е да се користи мерата која се добива со изразот (5.15).

$$M(x \in K_c) = \begin{cases} \dfrac{\sum_{i=1}^{N_v} e_i^c(x)}{\sum_{i=1}^{N_v}\sum_{s=1}^{N_c} e_i^s(x)} \cdot \dfrac{\sum_{i=1}^{N_v} e_i^c(x)}{N_v}, & \text{ако} \quad \sum_{i=1}^{N_v}\sum_{s=1}^{N_c} e_i^s(x) \neq 0 \\ 0, & \text{ако} \quad \sum_{i=1}^{N_v}\sum_{s=1}^{N_c} e_i^s(x) = 0 \end{cases} \tag{5.15}$$

Горните изрази за гласање и комбинирање на резултати од повеќе класификатори се базираат на одговорите на класификаторите дали дадениот примерок припаѓа или не на дадена класа, не земајќи ги при тоа во предвид грешките што ги прави секој од класификаторите при класификацијата на примероци од одредени класи. Грешките што ги прави секој од класификаторите се изразуваат преку т.н. вектор на конфузии (confusion vector). Конфузиониот вектор носи информација за тоа кои грешки најчесто ги прави дадената мрежа, односно примероците од кои симболички класи ги препознава како свои и во колкав број од случаите. Конфузиониот вектор има онолку елементи колку што различни класи може да се појават на влезот на класификаторот, при што секој елемент ја изразува соодветната веројатност примерокот кој го препознала дадената мрежа да припаѓа на дадена класа.

Во втората имплементација на комбинаторскиот систем отстрането е ограничувањето бројот на гласачите за сите симболички класи да биде ист и на секоја од ALN мрежите и е придружен конфузион вектор (Слика 5.22). Во ваков случај мерката која изразува дека даден примерок x е припадник на класата K_c е дадена со изразот (5.16).



Слика 5.22 Препознавање на знаци со употреба на различен број на гласачи по симболичка класа и тежинско гласање

$$M(x \in K_c) = \begin{cases} \dfrac{\sum_{s=1}^{N_c}\sum_{i=1}^{N_v^c} e_i^c(x)\cdot {}^c w_i^s}{\sum_{c=1}^{N_c}\sum_{s=1}^{N_c}\sum_{i=1}^{N_v^s} e_i^s(x)\cdot {}^c w_i^s} \cdot \dfrac{\sum_{s=1}^{N_c}\sum_{i=1}^{N_v^c} e_i^c(x)\cdot {}^c w_i^s}{\sum_{s=1}^{N_c}\sum_{i=1}^{N_v^s} {}^c w_i^s}, & \text{ако } \sum_{i=1}^{N_v}\sum_{s=1}^{N_c} e_i^s(x) \neq 0 \\ 0, & \text{ако } \sum_{i=1}^{N_v}\sum_{s=1}^{N_c} e_i^s(x) = 0 \end{cases} \quad (5.16)$$

каде:

N_c е вкупниот број на симболички класи;

N_v^c е бројот на гласачи за симболичката класа c;

$e_i^c(x)$ е бинарниот излез на i-тиот гласач за симболичката класа c кога на неговиот влез ќе се појави примерокот x;

${}^c w_i^s$ е c-тата вредноста на конфузиониот вектор придружен на i-тиот гласач за симболичката класа s која го изразува верувањето (belief factor) дека препознатиот примерок е припадник на класата c.

Вредностите на конфузиониот вектор на идеален класификатор наменет за препознавање на примероците од класата c би биле:

$$ {}^c w_i^c = 1; \;\; {}^s w_i^c = 0, \;\; s = 1, \ldots, N_c, s \neq c \quad (5.17)$$

За систем од идеални класификатори кои имаат еднаков број на гласачи по класа изразот (5.16) преминува во (5.15).

И во овој случај при препознавањето примерокот се упатува на препознавање на сите мрежи одговорни за препознавање на знаците од неговата shape класа. Секоја мрежа, ако го препознае примерокот, како излез дава низа од мерки од кои секоја го изразува верувањето дека примерокот е припадник на дадената класа, и тоа за секоја од симболичките класи во дадената shape класа. Овие мерки нормализирани и подредени во опаѓачки редослед ја формираат листата на кандидати со соодветната мерка на доверливост (confidence level) за секој кандидат. Првите неколку кандидати од листата со мерките на доверливост се проследуваат на контекстниот постпроцесор кој ја донесува конечната одлука за припадноста на примерокот. Доколку не се располага со постпроцесор, најмала грешка се прави доколку примерокот се прогласи за првиот во листата со кандидати (кандидатот со најголема мерка на доверливост).

Бидејќи при класификацијата примерокот се пушта на препознавање само на мрежите за симболичките класи од соодветната shape класа, евентуалната грешка во одредувањето на shape класата на знакот би значело и негово непрепознавање или погрешно препознавање, без тој воопшто да биде проследен на препознавање на соодветната мрежа. Затоа, за да се намали влијанието на грешките во одредувањето на shape класата, доколку примерокот не биде препознат во првиот круг, се упатува на препознавање од страна на мрежите одговорни за симболичките класи од shape класите најблиски до shape класата на примерокот, наречени алтернативни shape класи за дадениот примерок. Од дефиницијата на shape класите и од Слика 5.15 јасно е дека некои парови shape класи никогаш не може да се помешаат (како на пример . или , со '), но за оние кои лежат близу една до друга можно е примерок од една shape класа да биде погрешно предкласификуван како припадник на соседната shape класа. Затоа за секоја shape класа се одредени алтернативните класи кои треба да се проверат доколку примерокот не биде успешно препознат како ниту еден од знаците на shape класата која му била првично доделена. Алтернативните класи за секоја shape класа во редослед на веројатноста на замена се дадени на Слика 5.23.

Класа	Алтернативи
x	A y Q j –
A	X q y j !
y	Q j x A
Q	j A x y
j	Q ! A y
!	j A x

Класа	Алтернативи
.	нема
,	нема
'	нема
–	нема
:	! j
;	! j

Слика 5.23 Алтернативни класи за пооделните shape класи

За донесување одлука за прозивање и на гласачите за алтернативните shape класи, клучна е проценката дека примерокот не бил успешно препознат односно не е уверливо сврстен во ниту една од симболичките класи на shape класата која

му била доделена при предкласификацијата. Доколку по прозивањето на основното множество на гласачи мерката на доверливост на најдобриот кандидат не е доволно висока, се преминува на прозивање и на гласачите за првата од алтернативните shape класи, по што повторно се евалуираат мерките на доверливост, сега земајќи ги во предвид само резултатите од гласачите за алтернативната shape класа. Доколку и тука не се пронајде доволно добар кандидат се преоѓа на гласачите за втората алтернативна shape класа итн. Ако и по прозивањето на сите гласачи за сите алтернативни shape класи не се пронајде доволно добар примерок, сумарните резултати за мерките на доверливост добиени земајќи ги во предвид сите прозвани гласачи, се подредуваат и од нив се прави листата на кандидати. Доколку границата за мерката на доверливост која треба да ја надмине некој од кандидатите за да се смета како успешно препознат и да не се прозиваат другите гласачи е поставена на ниска вредност, веројатноста примерокот кој бил погрешно предкласификуван да биде и погрешно препознат ќе биде поголема. Ако пак оваа граница е поставена на висока вредност, премногу често ќе доаѓа до непотребно прозивање на алтернативните гласачи, со што ќе опаднат перформансите при препознавањето.

5.4.2 Обучување на класификаторите

Обучувањето (адаптацијата) на ALN мрежите се изведува со множеството за обука, кое ги содржи оригиналните (не скалирани) бинарни матрици на изолираните знаци за кои се познати нивната shape класа и симболичка класа. Shape класата за секој знак се одредува во процесот на предкласификација според формата на знакот, но симболичката класа мора де биде зададена однадвор, т.е. од некој кој веќе знае да го препознава примерокот. Во почетокот тоа е обично операторот, а потоа за препознавање може да се искористи и делумно обучениот систем и операторот само да ги воочува и исправа неговите грешки со што значително се забрзува подготовката на обучувачкото множество. Еднаш обучен, системот врши автоматско препознавање на примероците без интеракција од страна на корисникот.

Располагањето со што поголемо множество на примероци чии симболички класи се познати (наречено множество за обука на класификаторите) е многу важно бидејќи од ваквото множество се одвојуваат и обучувачкото множество (за ALN мрежите), и множеството со примероци за евалуација и нагодување на конфузионите вектори на ALN мрежите, и множеството за проценка на перформансите на класификаторите.

Подготовка на обучувачкото множество

За подготовка на обучувачкото множество беше изработена посебна програма која веќе изолираните знаци од некој скениран текст еден по еден ги прикажува на екранот и од операторот се бара да ја внесе соодветната симболичка класа на која припаѓа секој од знаците која се придружува кон сликата на секој знак. По оформувањето на едно обучувачко множество, кое можеби по обемот не е доволно за квалитетна обука на класификаторите, сепак се пристапува кон нивна обука. Вака (делумно) обучените класификатори потоа се употребуваат во процесот на понатамошното дооформување на обучувачкото множество за препознавање на примероците и предлагање на симболичката

класа која треба да се придружи на секој примерок. При ова работата на операторот се сведува на потврдување на предложената симболичка класа (за најголем дел од примероците доколку обучувањето било релативно успешно), воочување на погрешно препознатите и непрепознатите примероци и внесување на соодветната симболичка класа за нив. По делумното докомплетирање на обучувачкото множество може да се направи нова итерација на обучување на мрежите, со што понатамошното дополнување на обучувачкото множество ќе биде уште поефикасно од аспект на потребните корекции кои ќе треба да ги изврши операторот.

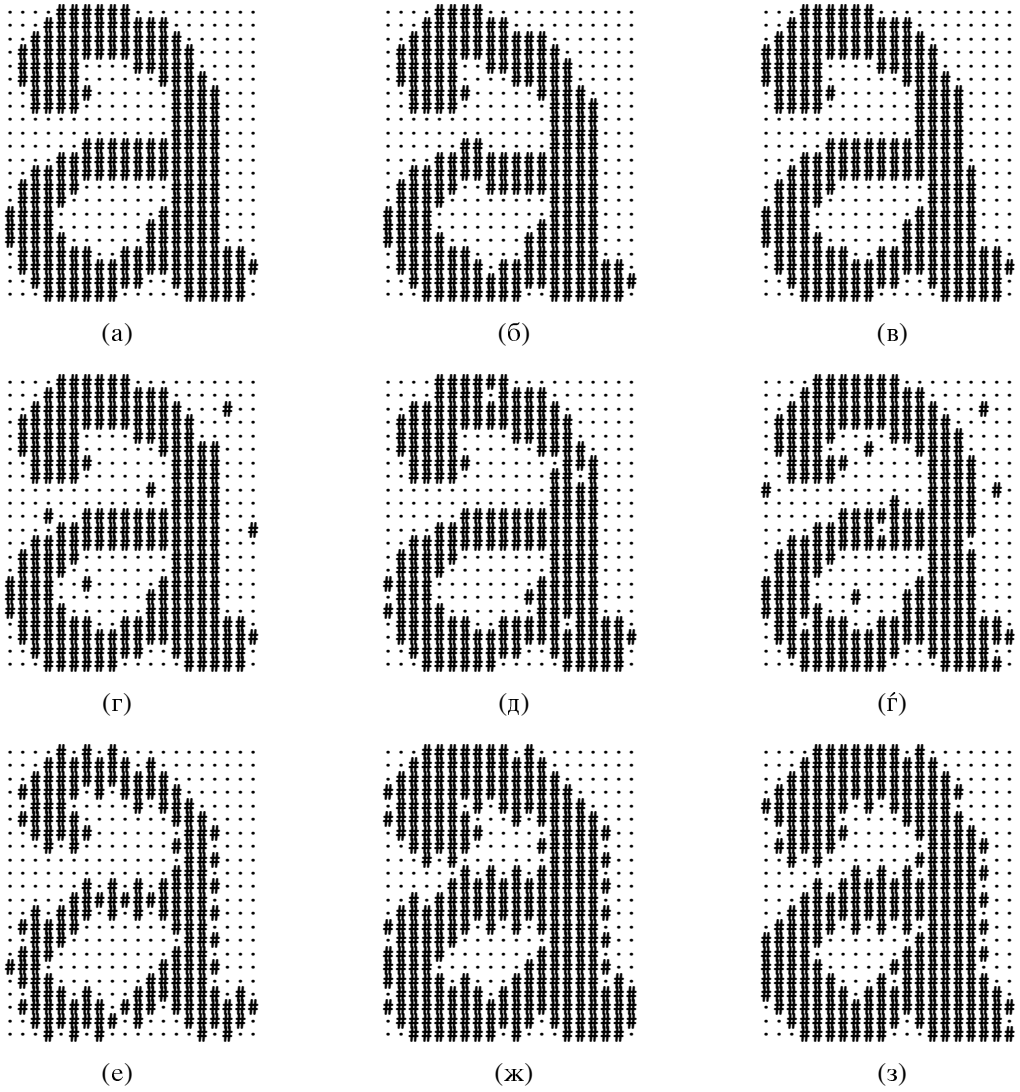
Во процесот на формирање на обучувачкото множество за секој од примероците постојат три можности:

1. да му се придружи соодветната симболичка класа;
2. примерокот да се отфрли (да не влезе во множеството за обука);
3. да се означи со специјална ознака за т.н симбол за отфрлање.

Примероците кои се премногу изобличени и при тоа премногу личат на знак од некоја друга симболичка класа се отфрлани за да не го преоптоваруваат со шум обучувачкото множество. Доколку обучувачкото множество содржи погрешно идентификувани примероци или примероци кои премногу личат на примерок кој е припадник на друга симболичка класа, обучувањето воопшто нема да успее. Примероците пак, кои претставуваат допрени или распаднати знаци не припаѓаат на ниту една од симболичките класи и доколку како такви не личат на некој од валидните знаци се означуваат како примероци за отфрлање односно примероци кои не треба да бидат препознати од ниту еден од класификаторите. Со ова се обучуваат класификаторите да ги отфрлаат погрешно сегментираните знаци, со што би се овозможила нивна детекција и евентуална поправка на сегментацијата.

Вештачки деформации на примероците

За квалитетна обука, нагодување на конфузионите вектори и евалуација на класификаторите е потребно што пообемно множество на примероци за кои е позната симболичката класа. Примероците одвоени од реални скенирани текстови (сé додека не се премногу изобличени) се најквалитетни од аспект на генерализацијата при обучувањето на класификаторите, бидејќи реално е да се очекува класификаторот да се сретне со слични примероци и нивни изобличувања при препознавањето на реални текстови. Сепак, подготовката на обучувачко множество од изолирани знаци од реални скенирани текстови бара ангажирање на значителни човечки ресурси. Затоа беше направен обид за автоматско генерирање на произволен број на делумно изобличени примероци од почетен идеален примерок со позната симболичка класа. За таа цел беше изработена специјална програма која за секој примерок од множеството со веќе класификувани примероци генерира нови примероци деформирајќи го влезниот примерок и/или додавајќи му шум.



Слика 5.24 Генерирани деформирани примероци: (а) оригинал, (б) поместен надоле од 8 до 20 колона, (в) поместен во лево од 0 до 12 редица; додаден шум (г) на позадина, (д) на знакот, (ѓ) и на позадина и на знакот

Деформациите кои се изведуваат на оригиналниот примерок вклучуваат едноставно поместување на дел од знакот по хоризонтала или по вертикала за еден пиксел (Слика 5.24 б,в). Ваквата деформација е типична деформација на знаците која настанува при корекцијата на закосеноста на сликата со текстот. Со оваа деформација може да се генерираат голем број на примероци кои се незначително изобличени верзии на оригиналниот примерок, но сепак различни од него. Друг вид на деформација која може да се изведе над оригиналниот примерок е негово истенчување или задебелување со назабување на рабовите (Слика 5.24 е-з). Освен со деформации, нови примероци може да се добијат и со додавање на шум на оригиналниот примерок. При тоа шумот што се додава може да се додаде само на позадината (расеани црни точки на позадината на знакот), само на знакот (расеани бели точки на знакот) или и на позадината и на знакот (Слика 5.24 г-ѓ).

Во програмата за генерирање деформирани примероци може да се нагласи за примероците од некои симболички или shape класи да не се генерираат деформирани примероци, како и да се ограничат минималните и максималните

димензии на матриците на знаците за кои ќе се генерираат деформирани примероци. Ограничувањето на димензиите за матриците има за цел да се исклучи генерирањето деформирани примероци за знаците со многу мали димензии бидејќи можат да доведат до значително изобличување на сликата на знакот. Кај знаците пак со премногу големи димензии, деформациите и додадениот шум најчесто се губат при нивното намалување при скалирањето. Исто така, се наведува бројот на деформирани примероци кои сакаме да бидат генерирани за секој оригинален примерок. При генерирањето на секој деформиран примерок начинот на деформација (правецот и насоката на поместување, позицијата во однос на која ќе се изврши и над која половина од знакот ќе се изврши поместувањето) случајно се избира или може да се наведе да се изврши генерирање на сите можни различни деформации. За генерирањето на примероци со додаден шум се специфицира бројот на примероци кои сакаме да бидат генерирани од секој оригинален примерок, нивото на шум кој се додава и начинот на додавање на шумот. Исто така можно е и вклучување и исклучување на генерирањето на истенчени и задебелени примероци.

За подготовка на обучувачкото множество беше изготвен текст кој ги содржи сите знаци кои треба да бидат распознавани од системот. Овој текст потоа беше "отпечатен" директно во TIFF датотека. Од идеалната неизобличена слика на ваквиот текст беа изолирани поодделните знаци и на секој од нив му беше придружена неговата симболичка класа, со што се доби множество од идеални недеформирани примероци. Од ова множество со програмата за генерирање деформации беа генерирани голем број на деформирани примероци. Исто така, беа издвоени знаци и од неколку скенирани реални текстови на кои им беа придружени соодветните симболички класи. Текстовите кои беа користени за изолирање на примероци беа земени од книги, часописи и дневен печат, како и изготвениот текст со сите знаци кој беше испечатен па потоа скениран и првите три генерации на фотокопии (фотокопија од фотокопија) на овој текст со зголемен и намален контраст при копирањето.

На овој начин беа добиени три множества со примероци:

1. идеално – добиено од изготвениот апсолутно недеформиран текст кое е релативно мало и го содржи секој знак во неколку различни големини и фонтови;
2. реално – во кое беа сместени примероците издвоени од реалните скенирани текстови и од изготвениот текст природно деформиран при копирањето и скенирањето кое броеше 42.630 примероци и
3. вештачко – добиено со генерирање на деформирани примероци од идеалното множество.

Додека бројот на примероците во првите две множества е ограничен, во третото множество можеме да генерираме произволен број на деформирани примероци. Множеството со реални примероци беше поделено на три дела од кои првиот содржеше околу 50% од примероците, вториот 30% и третиот преостанатите 20%. Множеството за обука на ALN мрежите кое броеше 90.244 примероци беше добиено од првиот дел од множеството со реални примероци на кое беа додадени вештачки генерирани деформирани примероци кои сочинуваа околу 75% од вкупниот број на примероци. Во составот на множеството за евалуација и нагодување на конфузионите вектори на ALN мрежите, влезе вториот дел од множеството со реални примероци на кое беа додадени вештачки

генерирани деформирани примероци кои сочинуваа околу 30% од вкупниот број на примероци. Последниот дел од множеството со реални примероци беше употребен за проценка на перформансите на класификаторите.

Обучување на мрежите

Бидејќи секоја ALN мрежа е одговорна за препознавање на примероци од само една симболичка класа (наречена основна симболичка класа за таа мрежа), таа се обучува да ги препознава примероците од нејзината симболичка класа, а да ги отфрла сите останати примероци кои ќе и бидат проследени. Ова овозможува секоја ALN мрежа да биде обучувана со за неа специфично конструирано обучувачко множество, оптимизирано за дадената мрежа. Обучувачките множества за секоја мрежа може директно да се извадат од главното обучувачко множество кое ги содржи примероците од сите симболички класи и всушност претставуваат негови подмножества. Во обучувачкото множество за секоја ALN мрежа освен примероците од нејзината основна класа влегуваат и примероците од други симболички класи кои таа мрежа треба да ги отфрла. Кои од другите симболички класи (за отфрлање) освен основната (за препознавање) ќе бидат вклучени во обучувачкото множество за дадена ALN мрежа се одредува на база на можните примероци од симболичките класи кои во процесот на препознавање можат да бидат проследени на препознавање на таа мрежа. Тоа се обично примероците од симболичките класи припаднички на истата shape класа и алтернативните shape класи на која припаѓа и основната симболичка класа за мрежата. Така на пример мрежата наменета да ги препознава примероците од симболичката класа **А**, мора да се обучи истовремено да ги отфрла останатите примероци од shape класата **A**, додека нема потреба да се обучува за отфрлање на знаците **.** и **,** бидејќи не постои можност во процесот на препознавање тие да бидат проследени на препознавање до оваа мрежа. Применувајќи ја оваа логика, може да се види дека за голем број на мрежи обучувачкото множество ќе биде заедничко, па ваквите мрежи може да се обучуваат во иста сесија. Згодно е при обучувањето на некои мрежи, од нивното обучувачко множество да се исклучат знаците од друга симболичка класа кои се многу слични на некој од знаците од основната симболичка класа за таа мрежа, а евентуалната субституција со токму таа класа може релативно лесно да се отстрани во процесот на контекстно постпроцесирање. Така на пример при обучувањето на мрежите за препознавање на **Р** и **У** од обучувачкото множество се отстрануваат примероците за **р** и **у**.

При обучувањето, сите примероци од обучувачкото множество за дадена ALN мрежа се скалираат на димензиите карактеристични за основната симболичка класа и придружени со ознаката која означува дали треба да бидат препознавани или отфрлани се проследуваат на мрежата. Костурот на алгоритамот за обука на ALN мрежите е даден на Слика 5.25.

```
load(primeroci_za_obuka);
skaliraj(primeroci_za_obuka);
za site simbolički klasi za dadena obučuvačka sesija
{
      za sekoj glasač na dadenata simbolička klasa
      {
            točnost=100%;
            do {
                  dimenzija=inicijalna_dimenzija;
                  do {
                        obid=0;
                        do {
                              mreža=konstruiraj_ALNmreža(dimenzija);
                              obučena=obuči(mreža, točnost);
                              obid++;
                        } while(! obučena && obid < max_broj_obidi);
                        dimenzija+=dim_čekor;
                  } while(! obučena && dimenzija < maksimalna_dimenzija);
                  točnost-=toc_čekor;
            } while(! obučena && točnost > minimalna_ točnost);
            if(obučena) save(mreža);
            else javi_greška();
      }
}
```

Слика 5.25 Костур на алгоритамот за обука на ALN мрежите

Колку што е поголема димензијата на ALN мрежата (бројот на нивоа односно длабочината), таа може да се адаптира да реализира посложена булова функција, односно да решава покомплексен проблем. Исто така, поради хеуристиката при случајното поврзување на вредностите на бинарниот вектор и неговите комплементи на влезовите на ALN мрежата, може да се случи поради непогодниот распоред на влезовите, во таа конфигурација мрежата да не може успешно да се обучи.

При обучувањето на ALN мрежите се почнува со конструкција на мрежа со одредени почетни димензии, и се прави обид таа да се обучи правилно да ги класификува сите примероци од обучувачкото множество. Доколку обучувањето не успее во првиот обид, се конструира нова мрежа (со што автоматски се иницијализира и реконфигурира) со истите параметри (број на нивоа) и обучувањето се повторува. Ако и после неколку вакви обиди обучувањето не е успешно, можеби проблемот е прекомплексен за дадената мрежа па циклусот се повторува за мрежа со поголем број на нивоа. Доколку и по неколку последователни зголемувања на мрежата, поточно ако таа достигне некоја однапред дефинирана максимална вредност, а обучувањето не успеало, се намалуваат барањата за точност во класификацијата (мрежата се смета за успешно обучена доколку правилно класификува одреден процент од сите влезни примероци) и целиот циклус се повторува. Ако и по неколку последователни намалувања на бараната точност, поточно кога таа ќе падне под некоја однапред зададена вредност, обучувањето се прекинува што најчесто значи дека обучувачкото множество има контрадикторни примероци.

Нагодување на конфузионите вектори на ALN мрежите

По обуката дури и ако ALN мрежата била успешно обучена на целото обучувачко множество не значи дека и апсолутно правилно ќе ги класификува сите нови примероци кои ќе и бидат проследени, односно реално е да се очекува дека таа ќе прави и извесни грешки. Доколку располагаме со дополнителна информација за тоа во која мерка и каков вид грешки прави мрежата ќе можеме поефикасно да ги искористиме одговорите кои таа ќе ги дава при препознавањето, односно да знаеме во која мерка можеме да се потпреме на сигурноста на нејзините одлуки во одредени случаи.

Грешките што ги прави секој обучен класификатор при класификувањето можат да се изразат преку т.н. вектор на конфузии:

$$\boldsymbol{v}_i^c = [n_1 \;\; n_2 \;\; \cdots \;\; n_j \;\; \cdots \;\; n_{N_c} \;\; n_\phi] \tag{5.18}$$

каде секое ${}^j v_i^c = n_j$ го претставува бројот на примероци од класата j кои биле препознати како припадници на класата c од страна на класификаторот e_i^c. ϕ ја претставува класата на примероци кои треба да бидат отфрлени од сите класификатори. Векторот на конфузии за даден обучен класификатор e_i^c може да се добие со употреба на e_i^c за класификација на примероците од множеството за тестирање. Ако во множеството за тестирање има N_j примероци од класата j, тогаш вкупниот број на примероци во множеството е:

$$N_T = N_\phi + \sum_{j=1}^{N_c} N_j \tag{5.19}$$

а вкупниот број на примероци кои класификаторот e_i^c ги препознал како свои е:

$$r_i^c = {}^\phi v_i^c + \sum_{j=1}^{N_c} {}^j v_i^c \tag{5.20}$$

Според ова веројатноста дека примерокот $x \in K_j, j=\phi,1,\ldots,N_c$ ако класификаторот e_i^c го препознал како свој може да се изрази преку условната веројатност:

$${}^j b_i^c = P(x \in K_j / e_i^c(x)) = \frac{{}^j v_i^c}{N_j} \cdot \frac{{}^j v_i^c}{r_i^c} \tag{5.21}$$

каде ${}^j v_i^c / N_j$ е веројатноста да се појави настанот e_i^c ако се појавил настанот $x \in K_j$, односно веројатноста класификаторот e_i^c да го препознае како свој примерокот $x \in K_j$, додека ${}^j v_i^c / r_i^c$ е веројатноста примерокот кој класификаторот e_i^c го препознал како свој да е од класата K_j. Првата веројатност ја моделира нерамномерната застапеност на примероците од различни класи во множеството за нагодување на конфузионите вектори, бидејќи нашето множество за нагодување има распределба на примероците по класите приближна на нивната распределба во реалноста, па примероците од некои класи се доста ретко застепени, додека други се презастапени. Втората веројатност директно ја изразува (не)успешноста на класификаторот правилно да ги класификува примероците од класата K_j, во рамките на сите примероци кои ги препознал како свои.

Од друга страна конфузиониот вектор може да се разбере како претходно знаење на експерт. По настанувањето на настанот $e_i^c(x)$, експертот ги соопштува своите верувања за (не)сигурноста дека примерокот x е припадник на секоја од класите K_j, $j=\phi,1,\ldots,N_c$ преку реален број ${}^j w_i^c(x)$ наречен мерка на веродостојност (belief value, mesure of belief). Вредностите ${}^j w_i^c(x)$ може да се разгледуваат и како тежини кои при комбинирањето на резултатите од класификаторите се додаваат на одлуката примерокот x да се распореди во класата j ако бил препознат од класификаторот e_i^c.

При нагодувањето на конфузионите вектори, секој од примероците од множеството за нагодување беше презентиран на сите ALN мрежи и за секоја мрежа беше бележено колку примероци од кои класи препознала како свои, односно добиени се векторите $\boldsymbol{v}_i^c$. Од овие вектори потоа според (5.19), (5.20) и (5.21) се пресметани вредностите за мерките на веродостојност. Со нормализација, од мерките на веродостојност конечно се добиваат соодветните тежини ${}^j w_i^c$ на конфузиониот вектор $\boldsymbol{w}_i^c$ придружен на ALN мрежата e_i^c според (5.22).

$$ {}^j w_i^c = \frac{{}^j b_i^c}{{}^\phi b_i^c + \sum_{j=1}^{N_c} {}^j b_i^c} \tag{5.22} $$

По обучувањето и нагодувањето на конфузионите вектори, ALN мрежите се сместуваат во датотека од која се вчитуваат од страна на програмата за препознавање на примероци. Комбинирањето на резултатите добиени од ALN мрежите при препознавањето автоматски динамички се нагодува во зависност од употребените гласачки мрежи за секоја класа, што овозможува потполно транспарентно додавање на нови гласачки мрежи за било која од класите. Бидејќи со зголемувањето на бројот на гласачи растат и глобалните перформанси на класификацијата, овозможено е да се додадат гласачи само на оние класи кои покажуваат најслаби перформанси во препознавањето.

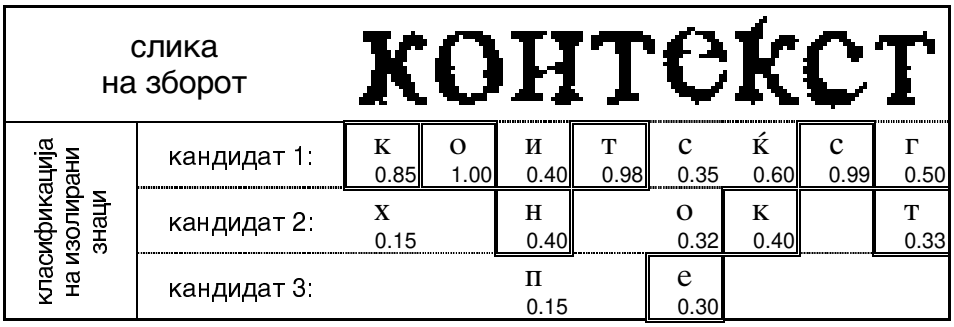
При проценката на перформансите на комбинацијата од обучени класификатори над множеството за проценка на перформанси кое броеше 8452 примероци, беше постигната точност од над 99.2% за првиот кандидат и над 99.9% за точен избор меѓу првите три кандидати.

5.5 Контекстно постпроцесирање

Контексното постпроцесирање има за цел да детектира и евентуално да исправи дел од грешките при препознавањето, како и да се обиде да ги "пополни празнините" односно врз база на контекстот да ги погоди непрепознатите знаци. Кај нашиот систем контекстното постпроцесирање е реализирано на ниво на збор со употреба на лексикон за македонскиот јазик. Лексиконот е реализиран како reTRIEval стебло за пребарување кое овозможува проверка на постоење на даден збор во лексиконот, како и снабдување на кандидати за непрепознатите букви во зборот.

5.5.1 Комбинирање на резултатите на класификацијата со употреба на контекстот на ниво на збор

Контекстниот постпроцесор врши обединување на резултатите добиени од класификаторот на изолирани знаци користејќи ги при тоа контекстните информации на ниво на збор и ја носи конечната одлука за припадноста на секој од изолираните знаци, односно е последната алка во препознавањето. На контекстниот постпроцесор му се проследуваат првите неколку кандидати од препознавањето на секој изолиран знак, сортирани според верувањето дека станува збор за токму тој знак (доверливоста) (Слика 5.26). Секој од кандидатите е проследен со квантитативната мерка за доверливоста на донесената одлука. За секоја знаковна позиција во рамките на еден збор може да постојат по неколку кандидат букви. Доколку некој од знаците воопшто не бил препознат, за таа позиција нема да постои ниту еден кандидат, знакот ќе биде регистриран како непрепознат, а постпроцесорот ќе направи обид да ја открие неговата вредност на база на контекстот.



Слика 5.26 Листа на кандидат букви за секоја знаковна позиција во рамките на зборот

Зборовите во кои нема непрепознати знаци се обработуваат на следниот начин: Се генерираат еден по еден зборовите кои може да се добијат со комбинирање на кандидатите на различните знаковни позиции и за секој од нив се проверува дали претставува валиден збор (дали го има во лексиконот). Ова се повторува сé додека не се наиде на валиден збор или додека не се исцрпат сите комбинации што може да се направат со комбинирање на кандидат буквите. Редоследот на генерирање на кандидатите е следниов: прво се генерира зборот составен од првите кандидати за секоја знаковна позиција во зборот. Ако овој збор не е пронајден во лексиконот се преминува на генерирање на зборовите кај кои само една од буквите не е прв кандидат, вршејќи замена на првиот кандидат со некој од останатите кандидати. Редоследот на земање на останатите кандидати е во опаѓачки редослед на доверливоста на кандидатите. Така во примерот од Слика 5.26 прво се заменува и со н, потоа ќ со к, па г со т итн., како што е прикажано на Слика 5.27. По исцрпување на сите кандидат зборови кои може да се добијат со замена на само една буква, се преминува на истовремена замена на две букви со некој од останатите кандидати на соодветната знаковна позиција, повторно во опаѓачки редослед на доверливостите на кандидатите. По ова се продолжува со истовремена замена на три букви итн., сé додека не се исцрпат сите можни комбинации или додека не се пронајде валиден збор. По пронаоѓањето на првиот валиден збор во секвенцата на генерирање кандидати тој се регистрира како препознат и се престанува со генерирање нови кандидати. Со ова практично е извршен избор на по еден од кандидатите за секоја знаковна

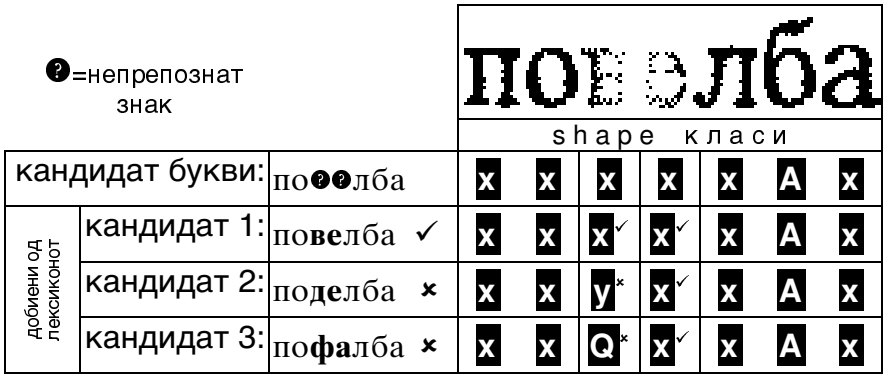
позиција во рамките на зборот на база на контекстот. Во случај и по исцрпување на сите кандидат зборови, меѓу нив да не се пронајде валиден збор, знаците се регистрираат како што се препознати од класификаторот, односно за секој знак се зема првиот кандидат. Генерално земено, може да се случи во комплетната листа на генерирани кандидат зборови да има повеќе од еден валиден збор. Во вакви случаи решението кој од валидните кандидати да се избере може да се донесе само со испитување на контекстот над нивото на изолирани зборови или со располагање со карактеристичните фреквенции на секој од зборовите во јазикот. Бидејќи во нашиот систем е имплементирана проверка на контекстот само до ниво на изолиран збор, во ваквите случаи ќе биде избран зборот кој попрв ќе биде генериран од листата на кандидат букви. Тоа значи дека ќе биде усвоен оној кандидат збор кој е генериран со помалку замени на првите кандидати со некој од останатите кандидати, а во рамките на кандидатите со ист број на заменети букви со оној чии заменети кандидати имаат повисоки доверливости.

к**о**н**т**с**ќ**с**г**	к**о**н**т**с**к**с**т**	к**о**н**т**о**к**с**т**
к**о**и**т**с**к**с**г**	к**о**н**т**о**к**с**г**	к**о**н**т**е**к**с**т**
к**о**и**т**с**ќ**с**т**	к**о**н**т**е**к**с**г**	**х**о**н**т**с**к**с**т**
к**о**и**т**о**ќ**с**г**	**х**о**н**т**с**к**с**г**	**х**о**н**т**о**к**с**г**
к**о**и**т**е**ќ**с**г**	к**о**н**т**о**ќ**с**т**	**х**о**н**т**е**к**с**г**
к**о**п**т**с**ќ**с**г**	к**о**н**т**е**ќ**с**т**	**х**о**н**т**о**ќ**с**т**
хо**и**т**с**ќ**с**г**	**х**о**н**т**с**ќ**с**т**	**х**о**н**т**е**ќ**с**т**
к**о**н**т**с**к**с**г**	**х**о**н**т**о**ќ**с**г**	к**о**п**т**о**к**с**т**
к**о**н**т**с**ќ**с**т**	**х**о**н**т**е**ќ**с**г**	**х**о**и**т**о**к**с**т**
к**о**н**т**о**ќ**с**г**	к**о**и**т**о**к**с**т**	к**о**п**т**е**к**с**т**
к**о**н**т**е**ќ**с**г**	к**о**и**т**е**к**с**т**	**х**о**и**т**е**к**с**т**
хо**н**т**с**ќ**с**г**	к**о**п**т**с**к**с**т**	**х**о**п**т**с**к**с**т**
к**о**и**т**с**к**с**т**	**х**о**и**т**с**к**с**т**	**х**о**п**т**о**к**с**г**
к**о**и**т**о**к**с**г**	к**о**п**т**о**к**с**г**	**х**о**п**т**е**к**с**г**
к**о**и**т**е**к**с**г**	**х**о**и**т**о**к**с**г**	**х**о**п**т**о**ќ**с**т**
к**о**п**т**с**к**с**г**	к**о**п**т**е**к**с**г**	**х**о**п**т**е**ќ**с**т**
хо**и**т**с**к**с**г**	**х**о**и**т**е**к**с**г**	**х**о**н**т**о**к**с**т**
к**о**и**т**о**ќ**с**т**	**х**о**п**т**с**к**с**т**	**х**о**н**т**е**к**с**т**
к**о**и**т**е**ќ**с**т**	к**о**п**т**о**ќ**с**т**	**х**о**п**т**о**к**с**т**
к**о**п**т**с**ќ**с**т**	**х**о**и**т**о**ќ**с**т**	**х**о**п**т**е**к**с**т**
хо**и**т**с**ќ**с**т**	к**о**п**т**е**ќ**с**т**	
к**о**п**т**о**ќ**с**г**	**х**о**и**т**е**ќ**с**т**	
хо**и**т**о**ќ**с**г**	**х**о**п**т**с**ќ**с**т**	
к**о**п**т**е**ќ**с**г**	**х**о**п**т**о**ќ**с**г**	
хо**и**т**е**ќ**с**г**	**х**о**п**т**е**ќ**с**г**	
хо**п**т**с**ќ**с**г**		

Слика 5.27 Генерирање на кандидат зборови со замена на 1, 2, 3 и 4 први кандидати со алтернативите (заменетите знаци се прикажани задебелено)

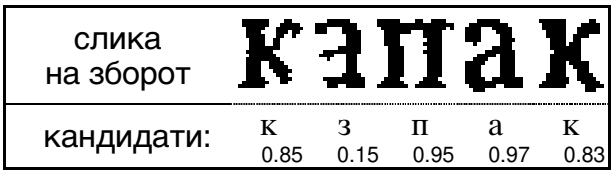
За зборовите кои во својот состав имаат и непрепознати букви (за одредени знаковни позиции не е доставен ниту еден кандидат) се прави обид да се пронајдат валидни букви со кои би се пополниле овие празнини така да зборот биде валиден. Во ваквите случаи би требало на местото на непрепознат знак да

се генерираат сите можни знаци за таа позиција како кандидати и да се провери во лексиконот дали постои таков збор. Нашиот лексикон е конструиран така да овозможува и пребарување низ него со делумно специфициран клуч, така да самиот како одговор ги врати сите можни погодоци, односно сите можни валидни зборови. Во случај да по ваквото пребарување се добие по само еден кандидат за секој непрепознат знак, односно постои само еден валиден кандидат збор во лексиконот тој и се усвојува. Во случајот да постојат повеќе валидни кандидат зборови, се чини дека за да се одлучи кој од нив да се усвои се потребни контекстни знаења над нивото на изолирани зборови или за карактеристичните фреквенции на секој од зборовите во јазикот. Сепак, иако за непрепознатите примероци не располагаме со ниту еден кандидат знак, веројатно располагаме со нивните shape класи. Ако се земе предвид дека предкласификацијата и на непрепознатите знаци често пати е правилно изведена, оваа информација може да се искористи при избирањето на збор од листата валидни зборови. Со споредување на shape класите на знаците на позициите на непрепознатите знаци во секој од кандидат зборовите со shape класите на непрепознатите знаци може да се редуцира листата на кандидат збориви и доколку ја сведеме на само еден збор, тој и да го усвоиме како валидна замена (Слика 5.28). Така на пример за двете непрепознати букви во 'по❓❓лба' по консултирање на лексиконот ќе бидат пронајдени три кандидати: 'по**ве**лба', 'по**де**лба' и 'по**фа**лба', од кои само првиот го задоволува условот третиот и четвртиот знак во него да се од shape класата **x**, од која се и непрепознатите знаци, па соодветно тој и ќе биде усвоен.



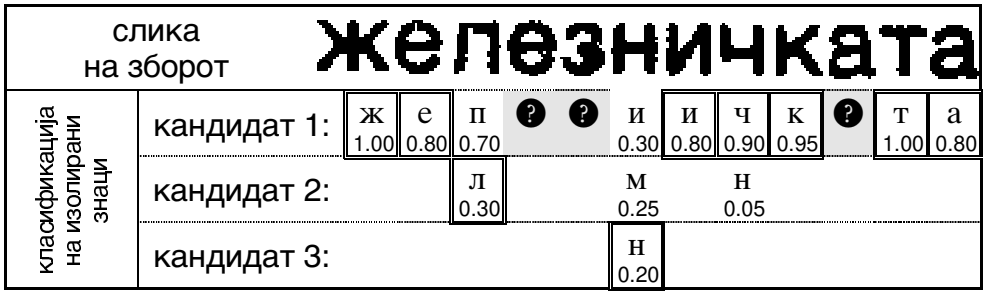
Слика 5.28 Барање и избор на алтернативи за непрепознати знаци

Доколку кај зборовите во кои за секоја знаковна позиција има по барем еден кандидат, по генерирањето на кандидат зборовите ниту еден од нив не е пронајден во лексиконот, може знаците на позициите на кои кандидатите имаат многу мала доверливост да се сметаат за непрепознати и за ваквиот збор да се спроведе барање низ лексиконот и избор од понудените кандидати. Така на пример за зборот препознат како 'кзпак' (Слика 5.29) првото 'а' поради деформацијата било прпознато како 'з' но со мала доверливост. По пребарувањето низ лексиконот како втората буква да не била воопшто позната, се добива единствениот кандидат 'капак'.



Слика 5.29 Третирање на знаците препознати со мала доверливост како непрепознати

Ако во зборот има и непрепознати знаци и знаци за кои се понудени повеќе кандидати (Слика 5.30) тогаш за секој збор генериран како комбинација од кандидатите букви се врши пребарување низ лексиконот со цел да се пронајде единствен валиден кандидат збор (Слика 5.31).



Слика 5.30 Збор со непрепознати знаци и со повеќе кандидати за некои знаковни позиции

Кандидат	Лексикон
жел❓❓иичк❓та	×
жеп❓❓иинк❓та	×
жеп❓❓мичк❓та	×
жеп❓❓ничк❓та	×
жел❓❓иинк❓та	×
жел❓❓мичк❓та	×
жел❓❓ничк❓та	✓ железничката

Слика 5.31 Генерирање кандидати и прегледување на лексиконот

Примероците препознати како специјалните знаци и цифри не се обработуваат од страна на контекстниот постпроцесор и тие се издаваат како што се препознати, освен во случаите кога за нив постојат и кандидати кои се букви. Буквите О, З и б (кои беа во исти симболички класи со цифрите 0, 3 и 6) доколку се најдат во збор во кој постојат други релативно уверливо препознати цифри се претвораат во соодветните цифри, а зборот се регистрира како број. Доколку во ваков збор се најде и некоја од буквите В или S, тие се променуваат во цифрите 8 и 5 соодветно.

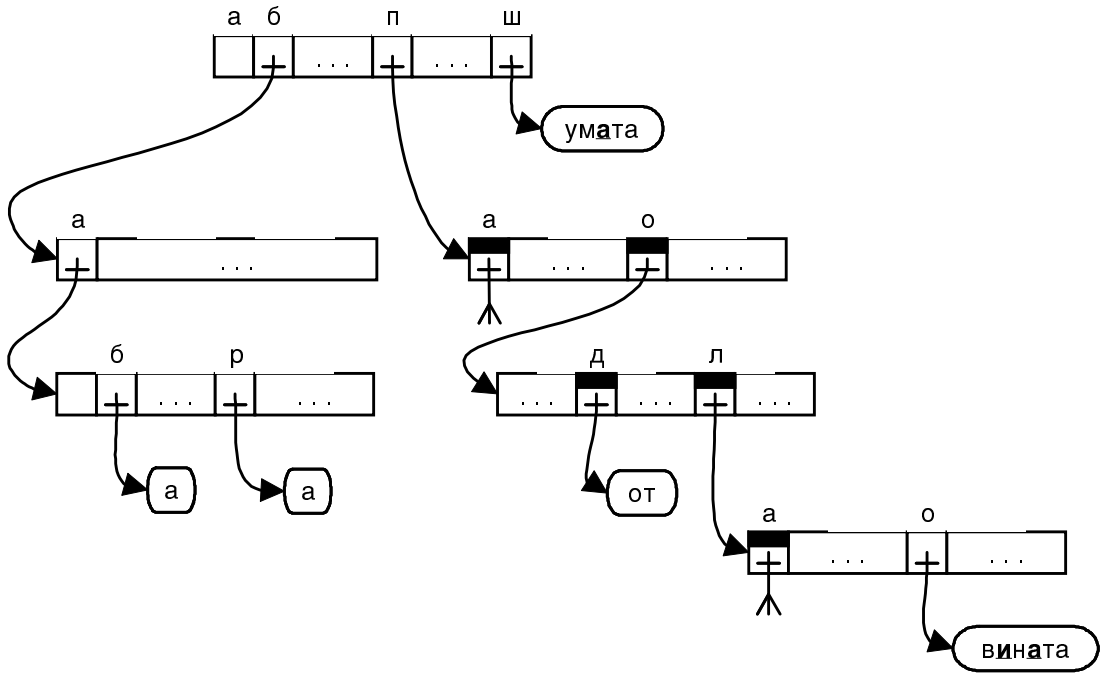
При контекстната проверка не се прави разлика меѓу малите и големите букви, но по регистрирањето на зборот, се врши промена на малите букви кои имаат ист облик како и големите и чиј примерок бил од shape класата **A** или **Q** во соодветните големи букви. Проблемот со правилното одредување на shape класите во редовите составен само од големи букви (наслови) може да се реши на следниот начин. Во ваквите редови нема да постојат знаци регистрирани како припадници на shape класите **A**, **Q** и **j**. Доколку по препознавањето некоја од буквите е уверливо препозната како буква од овие класи (што може лесно да се

случи при разгледувањето на алтернативните shape класи) значи дека базните линии во тој ред се погрешно проценети и треба соодветно да се променат shape класите на сите знаци од тој ред. Така на пример, доколку се случи некој знак првично предкласификуван во shape класата **x**, да биде препознат како цифра или како некој од знаците **А**, **Е**, **Б**, **J**, **Ф**, **Р**, **У** (знаци од shape класата **A** кои имаат различен облик од малите букви), или пак некој знак предкласификуван во shape класата **y**, да биде препознат како **ф**, означува дека е потребна корекција на shape класите. По регистрацијата и извршените корекции зборовите за запишуваат во излезната текстуална датотека.

5.5.2 Реализација на лексикон за македонскиот јазик

Контексниот постпроцесор е многу сличен на програмите за проверка на правопис (spell-checking) кои опфаќаат детекција на погрешно напишани зборови и предлагаат нивна замена во еден од листата валидни зборови кои се најслични на погрешниот. Сепак помеѓу проверката на правопис применета над текстови кои се внесувани од тастатура и оптички препознавани текстови постојат извесни разлики особено од аспект на понудувањето листа од валидни кандидати. Ова најмногу се должи на различната природа на грешките кои настануваат при внесувањето на текст и автоматското препознавање на печатени текстови. Така на пример чести грешки при внесувањето на текст се: невнесен знак, удвоен знак, вметнат знак, промена на местата на два соседни знаци и замена на знак, при што заменетиот или вметнатиот знак најчесто е некој од оние кои лежат на соседните тастери до оригиналниот знак. За разлика од ова кај оптичкото препознавање на текстови скоро единствена грешка која настанува е замената на еден знак со друг обично сличен по форма на оригиналниот знак, ако се изземат грешките при препознавањето на допрени и распаднати знаци. Меѓутоа и во едниот и во другиот случај потребна е база од валидни зборови со помош на која ќе може да се провери исправноста на даден збор. Ваквата база треба да ги содржи сите зборови кои може да се сретнат при проверката, па пожелно е да биде што покомплетна. Низ неа ќе треба да се врши што побрзо пребарување од типот дали даден збор постои во неа, односно ќе треба да биде индексирана по самите зборови кои можат да бидат со различни должини. Во пракса за реализација на компјутерски лексикони се употребуваат главно два пристапа за нивна организација и тоа со hash табели за брзо пребарување и како reTRIEval стебла.

ReTRIEval стеблата претставуваат јако разгранети стебла многу погодни за реализација на речници. Тие се состојат од јазли кои ги претставуваат буквите во валидните зборови. Секој јазел има онолку излезни гранки колку што букви има азбуката на јазикот чии зборови се чуваат во стеблото. Јазлите се поврзани во структура на разгрането стебло така да изминувањето на стеблото од коренот кон било кој лист секогаш дава валиден збор. Непостоечките зборови веднаш се пронаоѓаат бидејќи не претставуваат валидна патека низ стеблото или завршуваат на нетерминален јазел.



Слика 5.32 Пример за reTRIEval стебло

На Слика 5.32 е прикажано reTRIEval стебло во кое се сместени зборовите: баба, бара, па, по, под, подот, пол, пола, полови, половина, половината, шума и шумата. Заради заштеда на меморија уникатните краеви кои го завршуваат зборот со даден префикс (кои инаку би претставувале гранка со низа од јазли од кои секој има само еден наследник) се сместуваат во специјални јазли кои не овозможуваат понатамошно гранење. Терминалните јазли (јазлите со кои завршува валиден збор) на Слика 5.32 се означени со исцрнет горен дел, додека во специјалните јазли тоа е претставено со подвлечена буква. ReTRIEval стеблата овозможуваат брзо пребарување чија комплексност зависи само од должината на зборот.

За потребите на контекстниот постпроцесор во системот за оптичко препознавање македонски текстови беше реализиран и помал електронски лексикон за македонскиот јазик. Лексиконот е ралзиран во вид на reTRIEval стебло во кое се сместени околу 20.000 зборови. Поради непостоење на стандарден лексикон за македонскиот јазик во електронски читлива форма, бевме принудени во рамките на можностите самите да го реализираме. За таа цел беа издвоени зборовите од неколку подолги корегирани текстови кои ни беа на располагање како и околу 8.500 македонски зборови од паралелниот осумјазичен речник на CD-ROM [ER96]. За проверка при препознавање на неограничени текстови лексиконот треба да биде што покомплетен, затоа што кај малите речници многу често ќе се случува валиден збор да не биде пронајден во него.

Грешките кои што можат да се исправат со помош на лексиконите беа разгледани кај комбинирањето на резултатите од препознавањето. Меѓутоа некомплетноста на лексиконот може да предизвика и други грешки. Овие грешки предизвикуваат зборот за кои се доставени повеќе кандидати за одредени знаковни позиции, а за кои точно препознатиот знак секогаш е прв кандидат, доколку не постои во лексиконот да биде регистриран како потполно друг но

валиден збор кој постои во лексиконот. Доколку кандидат зборот формиран од првите кандидати иако валиден не постои во лексиконот, тој нема да биде веднаш регистриран туку ќе се премине на генерирање на кандидат зборови од кои некој можеби ќе биде и друг валиден збор кој ќе се најде во лексиконот. Доколку ниту еден од генерираните зборови не се наоѓа во лексиконот зборот сепак на крајот ќе биде валидно регистриран земајќи ги првите кандидати за секоја знаковна позиција. Генерално непостоењето на валиден збор во лексиконот може да услови замена на зборот со друг кој ќе биде генериран во процесот на генерирање на кандидат зборови.

слика на зборот	
кандидат 1:	н 0.55 а 0.95 п 0.95 о 0.97 н 0.63
кандидат 2:	п 0.15 к 0.23

Слика 5.33 Грешки поради некомплетност на лексиконот

На пример доколку во лексиконот не постои зборот ‘напон’, но постои зборот ‘папок’, при препознавањето на зборот од Слика 5.33 ќе дојде до негова замена со зборот ‘папок’. Грешката која настанува во овој случај не мора да се набљудува само како грешка затоа што некомплетниот лексикон не го содржел зборот ‘напон’, туку и како грешка затоа што лексиконот го содржел зборот ‘папок’, бидејќи ако ни тој не постоел во лексиконот, зборот ‘напон’ би бил правилно препознат. Грешките од овој тип би можеле делумно да се отстранат доколку се забрани замената на зборот добиен од првите кандидати со друг добиен со комбинирање на кандидати со значително помали доверливости, иако тој постои во лексиконот.

6 Заклучок

Во овој труд е презентиран систем за оптичко препознавање на печатен македонски кириличен текст. Ова е прв систем од овој вид кој овозможува препознавање на македонски текстови. Пред сé е наменет за помош на лица со оштетен вид и претставува дел од поголем систем за помош на лица со оштетен вид кој вклучува и претворање на текст во говор и печатење со браево писмо.

Системот овозможува потполно автоматско препознавање на печатени кирилични текстови. Врши автоматско детектирање на закосеноста на текстот, негово исправање, сегментирање и одредување на редоследот на читање. Препознавањето е изведено како препознавање на изолирани знаци. Пред препознавањето знаците се класификуваат според нивните димензии и вертикалниот простор кој го заземаат во редот со текст во 12 карактеристични класи. Овие класи се покажаа како доста независни од типот на писмото (фонтовите) и обезбедија разложување на множеството на можни знаци на подмножества од знаци со што се поедностави проблемот на класификацијата на знаците.

Класификацијата е изведена со употреба на адаптивни логички мрежи како класификатори. Адаптивните логички мрежи обезбедија брзина при обучувањето и препознавањето, а за постигање на подобри резултати употребени беа повеќе мрежи кои паралелно одлучуваат. Резултатите од препознавањето се комбинираат по пат на тежинско гласање, а конечната одлука се препушта на контекстниот постпроцесор, на кој му е проследена скоро целокупната информација добиена од класификаторите. Обезбеден е и транспарентен начин за додавање на нови гласачи и тоа само за одредени знаци со што може да се постигне подобрување на перформансите при препознавањето со директна интервенција на оние знаци за кои препознавањето е најслабо.

Контекстното постпроцесирање е изведено на ниво на изолирани зборови со употреба на лексикон за македонскиот јазик. Контексниот постпроцесор ја носи најдобрата одлука на база на контекстот и резултатите добиени од класификаторите. Познавањето на контекстот овозможува и детекција и исправање на одредени типови на грешки на класификаторите.

Бидејќи за македонскиот јазик досега не постои готова база на примероци на печатени знаци за обука на класификатори за оптичко препознавање на знаци, бевме принудени самите да ја формираме. При тоа беа рачно внесени симболите за околу 20.000 печатени знаци. Исто така поради непостоењето на стандарден лексикон за македонскиот јазик во електронски читлива форма, базата на зборови ја пополнивме со екстракција на зборови од корегирани текстови кои ни беа на располагање во електронска форма.

Со подобрување на процесот на сегментација би можело да се овозможи автоматско препознавање на документи со посложени формати, со слики и графици. Правците на понатамошните истражувања би требало да бидат насочени кон што поголема експлоатација на контекстот во процесот на препознавањето. Подигањето на анализата на контекстот над нивото на изолиран збор (колокации на зборови, анализа на фрази, синтаксна и семантичка анализа, …) може значително да ги подобри перформансите на препознавање особено кај неквалитетно печатените и оштетени текстови. Вакви анализи за македонскиот јазик досега не се спроведени, а не се истражувани ни модели за лексичка, синтаксна и семантичка анализа на македонскиот јазик.

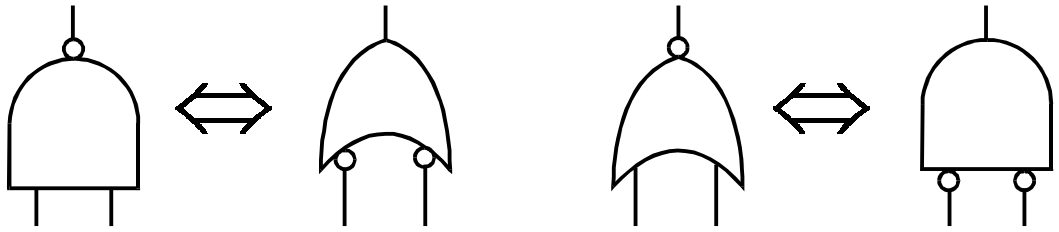
Прилог: Адаптивни логички мрежи

Адаптивната логичка мрежа (Adaptive Logic Network - ALN) е еден тип на вештачка невронска мрежа. Таа е специјален случај на познатиот повеќенивовски (повеќеслоен) перцептрон со простирање во една насока (multilayer perceptron (MLP) feedforward system). ALN е потполно бинарна мрежа – влезовите се бинарни, излезот е бинарен и целокупното процесирање во неа е потполно бинарно употребувајќи само логички операции (AND и OR) (или логички порти доколку е хардверски реализирана) и затоа е екстремно брза. За разлика од класичните аналогни невронски мрежи кои синтетизираат реални функции, ALN мрежите вршат синтеза на булови функции. Соодветно тие како резултат на својот излез даваат само една бинарна вредност.

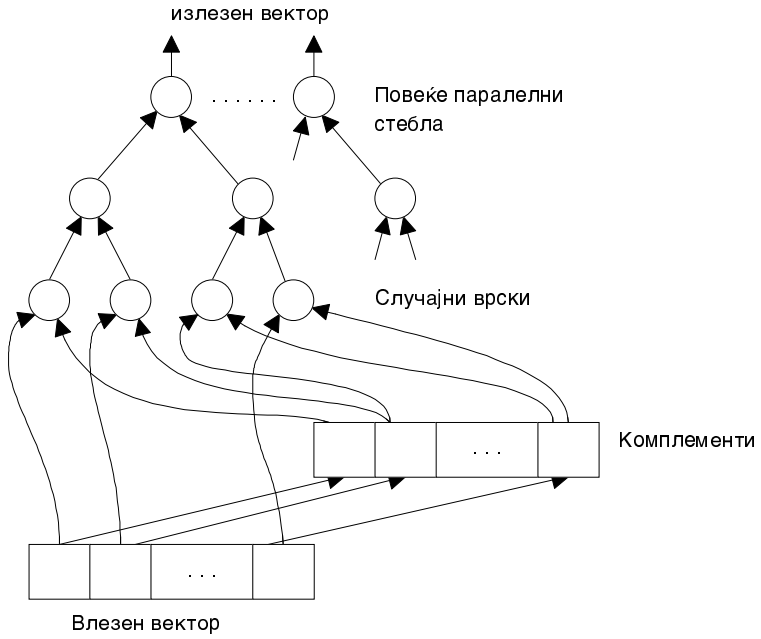
x_1	x_2	AND	LEFT	RIGHT	OR
0	0	0	0	0	0
0	1	0	0	1	1
1	0	0	1	0	1
1	1	1	1	1	1

Слика П1: Четирите функции во јазлите

ALN мрежите се организирани во вид на бинарно стебло од јазли кои претставуваат логички елементи кои вршат логичка операција над нивните влезови и резултатот го проследуваат на излезот. Листовите на ваквото стебло се поврзани на влезните променливи, а резултатот се добива на излезот од коренот на стеблото. Функциите $g(x,y)$ кои може да ги реализира секој од јазлите се: AND, OR, LEFT или RIGHT (Слика П1), т.е. $g(x,y)=x\cdot y$, $g(x,y)=x+y$, $g(x,y)=x$ и $g(x,y)=y$. Избраните функции ги претставуваат сите можни неконстантни растечки булови функции од две променливи. Растечките булови функции ја имаат особината да доколку било кој од нивните аргументи ја промени својата вредност од нула во единица, вредноста на резултатот не смее да се промени од единица на нула. Стеблото формирано од вакви јазли реализира исто така растечка булова функција. За да се отстрани ова ограничување, на влезот од стеблото се носат и инвертираните вредности на влезните променливи (Слика П3). Може да се покаже дека нема потреба од имплементирање на негации помеѓу елементите од стеблото, бидејќи негацијата од излезот на AND односно OR јазел може да се елиминира со нејзино преместување на неговите влезови и промена на јазелот во OR односно AND јазел (Слика П2). Итеративно истата операција може да се примени и на наследниците на променетиот јазел, сé додека единствените негации не останат само на влезовите на листовите.



Слика П2: Пропагација на негацијата низ логичките елементи



Слика П3: Шема на адаптивна логичка мрежа

Во споредба со повеќенивовскиот перцептрон чиј специјален случај е и самата, ALN мрежата има некои измени, од кои сите се поедноставувања. Сите "неврони" наречени адаптивни логички единици во стеблото на ALN мрежата имаат по точно два влеза x_1 и x_2, кои се бинарни (1 или 0). Секоја тежина на влезовите во секоја адаптивна логичка единица е одредена со само еден бит информација (односно врската или постои или не постои). Правилото за активација кое се користи кај адаптивните логички мрежи е едноставна прекинувачка функција (споредување со константа). Ако b_1 и b_2 се тежинските фактори на двата влеза во адаптивната логичка единица, тогаш таа го пресметува излезот, кој е единица ако и само ако:

$$(b_1 + 1)x_1 + (b_2 + 1)x_2 \geq 2$$

Четирите можни комбинации на b_1 и b_2 (00, 11, 10, 01) ги генерираат четирите булови функции од две варијабли: AND, OR, LEFT и RIGHT. Функциите LEFT и RIGHT ефективно служат само да обезбедат флексибилност во претставата на поврзувањето, а по завршувањето на обучувањето на стеблото тие исчезнуваат.

Доволно голема ALN мрежа може да реализира произволна булова функција, но тие најчесто се употребуваат за реализација на само делумно

дефинирани булови функции најчесто зададени во вид на множество за обучување од парови влезни вектори и посакуваниот излез. Конструкцијата на ALN мрежа која ќе реализира некоја делумно дефинирана булова функција, е наречена и процес на обука или адаптација на ALN стеблото. Во процесот на обука се поаѓа од ALN мрежа чии јазли се случајно иницијализирани на една од четирите можни функции. На ваквата мрежа и се презентираат во случаен редослед влезните вектори од обучувачкото множество. За секој вектор се пресметува функцијата која ја реализира стеблото, се споредува со бараниот излез и интерните состојби на некои од јазлите се менуваат. Ова резултира во доделување на функции на јазлите што му овозможува на стеблото да ги апроксимира саканите излези специфицирани со податоците за обучување. Обуката престанува во моментот кога функцијата реализирана од страна на стеблото ќе ги задоволи барањата.

Обучувањето на ALN мрежата се одвива по алгоритам сличен на back-prpagation алгоритамот и практично претставува негова бинарна верзија – сигналот за грешка е бинарна вредност и се пропагира наназад како бинарна вредност. Обучувањето се темели на особината на строго растечките функции кои ги реализираат јазлите и се состои во одредувањето на јазлите кои имаат *вистинска* и *хеуристичка* одговорност (true responsibility, heuristic responsibility) за излезот од мрежата и нивна адаптација [AG79]. За еден јазел се вели дека е вистински одговорен доколку промената на неговиот излез предизвика и промена на излезот од целото стебло, под услов сите останати сигнали во мрежата освен оние на патеката од него до коренот на стеблото да не се променат. Овој концепт може рекурзивно да се прошири почнувајќи од коренот на стеблото со испитување на функциите во јазлите и нивните влезови. Обучувањето односно измените кои се вршат во мрежата за време на нејзиното тренирање, се базираат на идејата дека доколку едниот влез во AND (односно OR) јазелот е 0 (односно 1), тогаш не е важно што се случува со јазлите од подстеблото поврзано на другиот влез. На пример ако едниот од влезовите во AND јазел е 0, тогаш вредноста на другиот влез не е битна за излезот од јазелот, па соодветно наследникот поврзан на тој влез не е вистински одговорен, односно вистинската одговорност се пренесува на другиот наследник. Аналогно ако влезот во AND јазелот е 1, тогаш другиот наследник е вистински одговорен. Слично резонирање може да се спроведе и за OR јазлите.

Според ова едноставно правило, јазелот е одговорен само доколку промената во неговиот излез би предизвикала и промена во излезот на целата мрежа. Ова е слично на backpropagation алгоритамот само што вредностите кои се пренесуваат наназад се бинарни, односно или 0 или 1. Фактот дека сите функции кои можат да се доделат на јазлите се растечки, условуваат постоење на позитивна корелација меѓу излезот од било кој јазел и излезот на целата мрежа, од каде може де се најде и правецот на промените кои треба да се направат во мрежата.

Обучувањето се одвива преку евалуација и адаптирање на функциите во јазлите, односно одлучување која од функциите AND, OR, LEFT или RIGHT треба да се додели на јазелот кој се обработува. Бидејќи сите четири функции кои може да се доделат на даден јазел го имаат својството да за влезови (0, 0) даваат излез 0 и за влезови (1, 1) даваат излез 1, единствено што останува е да се одреди кој излез треба да го дава јазелот во случајот на (1, 0) и (0, 1) влезни парови. За да

се одреди функцијата на јазелот за време на обучувањето во секој јазел постојат два бидирекциони ограничени бројачи, од кои едниот ги брои појавите на (1, 0), а другиот на (0, 1) влезни парови соодветно. Секој од бројачите се инкрементира или декрементира зависно од парот кој се појавил на влезовите и од бараниот излез од мрежата (целата мрежа, бидејќи како што споменавме функциите во јазлите се растечки па постои позитивна корелација меѓу излезот од целата мрежа и било кој елемент поединечно). Вредностите на овие два бројачи ја одредуваат функцијата која ќе биде доделена на јазелот. Ако и двата бројача се под нивните средишни точки тогаш се доделува функцијата OR, ако се над функцијата AND, а ако се на спротивните страни од средишните точки тогаш се доделуваат функциите LEFT или RIGHT.

Во зависност од доделената функција на јазелот и вредноста на влезовите во него се одредува на кое подстебло ќе се додели хеуристичката одговорност. Kај AND и OR јазлите ако на пример десниот влез не е еднаков на саканиот излез од мрежата, левото подстебло се прогласува за хеуристички одговорно. Логиката за ова е левото подстебло да се направи одговорно доколку во десното има грешка, бидејќи десното подстебло можеби не може да ја поправи грешката (можеби тоа е лист). Истото важи и за левото подстебло доколку десниот влез не е еднаков на саканиот излез од мрежата. Ако функцијата на јазелот не е ниту AND ниту OR (значи е или LEFT или RIGHT) тогаш хеуристички одговорно се прави исклученото подстебло. Со ова се прави обид подстеблата кои се отсечени со LEFT или RIGHT функциите, корисно да се искористат. Ако вака отсеченото подстебло во понатамошната обука стане доволно добро, функцијата на јазелот може да се промени во AND или OR, подстеблото да се вклучи во мрежата и да има ефективна улога во одлучувањето. Конечно ако левиот влез е 0 во јазел кој има OR функција или е 1 во јазел кој има AND функција, односно доколку е овозможено простирањето на десниот влез нагоре, тогаш десното подстебло е хеуристички одговорно.

Од изложеното може да се забележи дека левиот влез ја контролира одговорноста на десното подстебло и обратно. Од овде произлегува дека секој јазел може да влијае на секој друг јазел во подстеблото. Адаптирањето се одвива рекурзивно, јазел по јазел низ целото стебло. Ако стеблото дава погрешен излез по првиот циклус на адаптација се минува на втор итн. По завршувањето на обучувањето на мрежата, од неа можат да се отстранат подстеблата кои се исклучени со LEFT и RIGHT функциите, и тоа да се конвертира во стебло на одлучување кое побргу евалуира и зафаќа доста помалку меморија.

По обучувањето адаптивната логичка мрежа има вградена способност да генерализира, т.е. да дава соодветен излез и за други можни влезни вектори кои не биле презентирани на мрежата за време на обучувањето. Оваа особина се должи на својството на ALN мрежите, поточно генерираните функции кои тие ги реализираат за влезни вектори кои се блиски меѓу себе по хаминговите растојанија да даваат ист излез. Генерализацијата делумно се должи на својството на *неосетливост* (insensitivity) карактеристично за ALN мрежите. Неосетливоста се огледува во својството малите промени на влезниот вектор (мерени со хамингова метрика) да условуваат многу мала веројатност на промена на излезот. Промената на вредноста на влезот на еден јазел (од 1 во 0 или обратно) може но не мора да го промени и излезот од јазелот, а во просек веројатноста за промена на излезот доколку еден од влезовите се промени е 0.5.

Соодветно веројатноста дека излезот од јазелот на следното ниво ќе се промени е 0.25. За мрежа со N нивоа, промената на еден бит на нивото на листовите (влезовите) резултира во веројатност дека ќе дојде до промена на излезот од мрежата од само $1/2^N$, што за мрежа со 10 нивоа изнесува околу 0.00098. Од неосетливоста на ALN мрежите директно произлегува нивната извонредна отпорност на шум.

Адаптивните логички мрежи се мошне погодни за имплементација во хардвер, бидејќи се состојат од хиерархиски поврзано стебло составено само од AND и OR порти кои лесно се имплементираат и многу бргу евалуираат. Дури и ако софтверски се емулираат, адаптивните логички мрежи се побрзи од другите модели на невронски мрежи (на пример од MLP). Забрзувањето се должи најмногу на особината наречена *штедлива* или мрзлива евалуација (parsimonious evaluation) карактеристична за ALN мрежите. Штедливата евалуација се должи на фактот што кога ќе се пресмета едниот влез во јазелот чија функција е AND (или OR) и ако тој е 0 (или 1) тогаш не мора да се евалуира целото друго подстебло бидејќи и без тоа резултататот е веќе познат. Забрзувањата кои се добиваат со ова се во просек $(4/3)^N/2$, што за мрежа со N=10 нивоа изнесува околу 8.9, а за мрежа со N=20 нивоа дури 157 [ADL91]. Слично за пресметување на излезот на мрежата во просек ќе биде потребно испитување на само 1.5^N од можните 2^N влезни вредности, што за мрежа со N=20 нивоа значи испитување на само околу 3325 од можните преку милион влезови.

Особено важна карактеристика е тоа што со зголемувањето на проблемот комплексноста на стеблото расте релативно бавно (за два пати повеќе влезови ќе имаме само за едно ниво поголемо стебло) во однос на да речеме MLP. ALN мрежите се исто така и значително побрзи при обучувањето од MLP. Друга предност на адаптивните логички мрежи е што можеме при обуката да им ги проследиме сите податоци со кои располагаме без да вршиме селекција само на потребните за донесување на одлука, бидејќи алгоритамот за учење тоа сам ќе го стори (ќе ги исклучи стеблата со влезовите кои не влијаат на одлуката). Самата структура и начин на адаптација на ALN мрежите вклучува автоматска детекција и исклучување на непотребните делови од мрежата (преку LEFT и RIGHT функциите во јазлите), па нема потреба од посебно кастрење (pruning) на обучената мрежа.

Адаптивните логички мрежи иако во основа се бинарни можат да се применат и на аналогни податоци, доколку тие претходно се дигитализираат, и претстават во форма на бинарен вектор. Дигиталните излези на крајот може повторно се аналогизираат. Поради вградената хеуристика при иницијализацијата (случајниот избор и распоред на влезовите кои ќе се поврзат на влезот од стеблото), ALN мрежите се особено погодни за комбинирање и без специфичната поделба на обучувачкото множество при специфичното обучување на класификатори наменети за комбинирање.

Литература

[ADG95] Hadar I. Avi-Itzhak, Thanh A. Diep, Harry Garland, "High Accuracy Optical Character Recognition Using Neural Networks with Centroid Dithering", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 2, pp. 218-224, February 1995.

[ADL+91] William W. Armstrong, Andrew Dwelly, Jiandong Liang, Dekang Lin, Scott Reynolds, "Some Results concerning Adaptive Logic Networks", University of Alberta, Edmonton, Alberta, Canada T6G 2H1, September 1991.
[ftp://ftp.cs.ualberta.ca/pub/atree/atree2/atree2.ps.Z]

[AG75] W. Armstrong, G. Godbout, "Properties of Binary Trees of Flexible Elements Useful in Pattern Recognition", *IEEE Int. Conference on Cybernetics & Society*, pp. 447-449, San Francisco, 1975.

[AG79] William W. Armstrong, Jan Gecsei, "Adaptation Algorithms for Binary Tree Networks", *IEEE Transactions on Sytems, Man and Cybernetics*, Vol. 9, No. 5, May 1979.

[Ali96] Kamal Mahmood Ali, "On Explaining Degree of Error Reduction due to Combining Multiple Decision Trees", *Proceedings of the National Conference on Artificial Intelligence*, (workshop on Integrating Multiple Learned Models), pp. 1-7, Portland, Oregon, August 1996.
[http://www.cs.fit.edu/~imlm/papers/ali.ps]

[AP93] Horacio Ayestaran, Richard Prager, "The Logical Gates Growing Network", Cambridge University Engineering Department Technical Report CUED/F-INFENG/TR137, Trumpington Street, CB2 1PZ, July 1993.

[Atr92] Atree 2.7 simulator on-line Help, 1992.
[ftp://ftp.cs.ualberta.ca/pub/atree/atre27.exe]

[Bai87] Henry S. Baird, "The Skew Angle of Printed Documents", *40th Conference and Symposium on Hybrid Imaging Systems - SPSE*, pp. 21-24, Rochester, New York, May 1987.

[Bai88] Henry S. Baird, "Feature Identification for Hybrid Structural/Statistical Pattern Classification", *Computer Vision, Graphics and Image Processing*, Vol. 42, pp. 318-333, 1988.

[Bai93] Henry S. Baird, "Document Image Defect Models and their Uses", *Proceedings of the International Conference on Document Analysis and Recognition*, pp. 62-67, Tsukuba Science City, Japan, October 1993.

[BC+92] Luca Boatto, Vincenzo Consorti, Monica Del Buono, Silvano Di Zenzo, Vincenzo Eramo, Allessandra Esposito, Francesco Melcarne, Marco Meucci, Andrea Morelli, Marco Mosciatti, Stefano Scarci, Marco Tucci, "An Interpretation System for Land Register Maps", *IEEE Computer Magazine*, Vol. 25, No. 7, pp. 25-33, July 1992.

[BC95] Lionel Beaurepaire, Kacem Chehdi, "Iterative Filtering Method of Impulsive Noise with Edge Preservation", *Proceedings of IASTED International Conference: Signal and Image Processing - SIP-95*, Las Vegas, Nevada, USA, November 20-23, 1995.

[BCG+93] J. L. Blue, G. T. Candela, P. J. Grother, Rama Chellappa, C. L. Wilson, "Evaluation of Pattern Classifiers for Fingerprint and OCR Applications", CS-TR-3162, University of Maryland, October 1993.
[http://www.cs.umd.edu/Server/TR/UMCP-CSD:CS-TR-3162/Body?format=postscript]

[BCG+94] J. L. Blue, G. T. Candela, P. J. Grother, R. Chellappa, C. L. Wilson, J. D. Blue, "Evaluation of Pattern Classifiers for Fingerprint and OCR Application", *Pattern Recognition*, Vol. 27, pp. 485-501, 1994.
[ftp://sequoyah.ncsl.nist.gov/pub/papers_preprints/eval_classifiers.ps.Z]

[Bis95] Christopher M. Bishop, *Neural Networks for Pattern Recognition*, Clarendon Press, London, 1995.

[BJF90] H. Baird, S. Jones, S. Fortune, "Image Segmentation by Shape-Directed Covers", *Proceedings of 10th Internacional Conference on Pattern Recognition*, pp. 820-825, Atlantic City, New Jersey, June 1990.

[BKP86] H. S. Baird, S. Kahan, T. Pavlidis, "Components in an Omnifont Page Reader", *Proceedings of theEighth International Conference on Pattern Recognition*, pp. 344–348, Paris, 1986.

[BM80] Carlo Braccini, Giuseppe Marino, "Fast Geometrical Manipulations of Digital Images", *Computer Graphics and Image Processing*, Vol. 13, pp. 127-141, 1980.

[BN94] Henry S. Baird, George Nagy, "Self-Correcting 100-Font Classifier", *Proceedings of the SPIE*, Vol. 2181, Document Recognition, 1994.

[Boj89] Zoran S. Bojković, *Digitalna obrada slike*, Naučna Knjiga, Beograd, 1989.

[Bre93] Leen Breure, “Introduction in Optical Character Recognition in the Historical Discipline”, *Proceedings of an International Workshop* organized by Netherlands Historical Data archive and Nijmegen Institute for Cognition and Information, 10-12 June 1993, Max-Planck-Institut für Geschichte, Göttingen 1993.

[BS96] Robert R. Bailey, Mandyam Srinath, “Ortogonal Moment Features for Use With Parametric and Non-Parametric Classifiers”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 18, No. 4, pp. 389-399, April 1996.

[BT90] Henry S. Baird, K. Tompson, “Reading Chess”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, No. 6, pp. 552-559, June 1990.

[CAS93] J. Cao, M. Ahmadi, M. Shridhar, “Character Recognition Using Neural Based Extractor and Classifier”, *Proceedings of IEEE International Symposium on Circuits and Systems*, Vol. 4, pp. 2442-2445, 1993.

[CC95] P. C. Chee, C. Chan, “Random Noise Removal of Binary Text Images Using Seed Filling”, *Proceedings of IASTED International Conference: Signal and Image Processing - SIP-95*, Las Vegas, Nevada, USA, November 20-23, 1995.

[CL96] Richard G. Casey, Eric Lecolinet, “A Survey of Methods and Stretedies in Character Segmentation”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 18, No. 7, pp. 690-706, July 1996.

[CN82] R. G. Casey, G. Nagy, “Recursive Segmentation and Classification of Composite Patterns”, *Poceedings of the Sixth International Conference on Pattern Recognition*, p. 1023, 1982.

[CS80] Edwin Catmull, Alvy Ray Smith, “3-D Transformations of Images in Scanline Order”, Computer Graphics, (*SIGGRAPH '80 Proceedings*), Vol. 14, No. 3, pp. 279-285, July 1980.

[CS96] P. Chan, S. Stolfo, “Scaling Learning by Meta-Learning Over Disjoint and Partially Replicated Data”, in FLAIRS-96, 1996.
[http://www.cs.columbia.edu/~sal/hpapers/metalrep.ps]

[CST+92] L. P. Cordella, C. De Stefano, F. Tortorella, M. Vento, “Improving Character Recognition Rate by a Multi-Net Neural Classifier”, *Proceedings of the International Conference on Pattern Recognition*, Vol. 11, No. 2, pp. 615-618, IEEE, 1992.

[Dav90] E. R. Davies, Machine Vision: Theory, Algorithms, Practicalities, Academic Press Ltd., 1990.

[DBH+92] Andreas Dengel, Rainer Bleisinger, Rainer Hoch, Frank Fein, Frank Hönes, “From Paper to Office Document Standard Representation”, *IEEE Computer Magazine*, Vol. 25, No. 7, pp. 63-67, July 1992.

[DD95] Iwona Dabowska, Dorota Depowska, "Character Segmentation Method in Proportional Character Recognition", *Proceedings of 17th International Conference ITI '95*, pp. 241-246, Pula, Croatia, June 1995.

[DDS+96] Dov Dori, David Doermann, Christian Shin, Robert Haralick, Ihsin Phillips, Mitchell Buchman, David Ross, Chapter XX: "The Representation of Document Structure: a Generic Object-Process Analysis", *Handbook on Optical Character Recognition and Document Image Analysis*, Eds. P. S. P. Wang and H. Bunke, World Scientific Publishing Company, 1996.
[ftp://documents.cfar.umd.edu/pub/contrib/papers/UMD/1996/96-0001/93-0001.ps.gz]

[Den90] Andreas Dengel, "A Step Toward Understanding Paper Documents", Research Report RR-90-08, Deutsches Forschungszentrum für Künstliche Intelligenz GmbH, Postfach 2080, 67608 Kaiserlautern, FRG, 1990.
[ftp://ftp.dfki.uni-kl.de/Publications/ResearchReports/90/RR-90-08.ps.gz]

[Doe93] David S. Doermann, "*Document Image Understanding: Integrating Recovery and Interpretation*", PhD Disertation, CS-TR-3056, Document Processing Group, Center for Automation Research, University of Maryland, January 1993.
[ftp://documents.cfar.umd.edu/pub/contrib/papers/UMD/1993/93-0001/93-0001.ps.gz]

[Doe94] David Doermann, "Document Understanding Research At Maryland", *ARPA Image Understanding Workshop*, pp. 817-826, 1994.
[ftp://documents.cfar.umd.edu/pub/contrib/papers/UMD/1994/94-0002/94-0002.ps.gz]

[ECD94] Kamran Etemad, Rama Chelappa, David Doermann, "Document Page Segmentation by Integrating Distributed Soft Decisions", *International Conference on Neural Networks*, 1994.
[ftp://documents.cfar.umd.edu/pub/contrib/papers/UMD/1994/94-0003/94-0003.ps.gz]

[EDC94] Kamran Etemad, David Doermann, Rama Chelappa, "Page Segmentation Using Decision Integration and Wavelet Packet Basis", *International Conference on Pattern Recognition*, 1994.
[ftp://documents.cfar.umd.edu/pub/contrib/papers/UMD/1994/94-0004/94-0004.ps.gz]

[EDC95] Kamran Etemad, David Doermann, Rama Chelappa, "Multiscale Document Page Segmentation Using Soft Decision Integration", Submitted to IEEE Transactions on Pattern Aanlysis and Machine Intelligence, Document Processing Group, Center for Automation Research, University of Maryland, 1995.
[ftp://documents.cfar.umd.edu/pub/contrib/papers/UMD/1995/95-0007/95-0007.ps.gz]

[Elm96] J. Elms, "*The representation and recognition of text using hidden Markov models*", PhD Disertation, University of Surrey, Guildford, Surreu GU2 5XH, U. K., April 1996.
[ftp://ftp.ee.surrey.ac.uk/pub/vision/papers/elms-phd96.ps.Z]

[ENC96] Microsoft ENCARTA encyclopedia 1996 CD-ROM (writing systems), Microsoft, 1996.

[ER96] Европски речник CD-ROM, Založba mladinska knjiga, Ljubljana, 1996.

[FCS96] David W. Fan, Philip K. Chan, Salvatore J. Stolfo, "A Comparative Evaluation of Combiner and Stacked Generalization", *Proceedings of the National Conference on Artificial Intelligence*, (workshop on Integrating Multiple Learned Models), pp. 40-46, Portland, Oregon, August 1996.
[http://www.cs.fit.edu/~imlm/papers/fan.ps]

[FHD90] J. L. Fisher, S. C. Hinds, D. P. D'Amato, "A Rule-Based System for Document Image-Segmentation", *Proceedings of 10th Internacional Conference on Pattern Recognition*, pp. 567-572, Atlantic City, New Jersey, June 1990.

[FK88] Lloyd A. Fletcher, Rangachar Kasturi, "A Robust Algorithm for Text String Separation from Mixed Text/Graphics Images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 10, No. 6, pp. 910-918, November 1988.

[FKS96] Shinichi Fujitsu, Taiho Kanaoka, Josef Skrzypek, "Performance of Back Propagation Algorithm Using Dissimilarities between Target Patterns", *Proceedings of the Fourteenth IASTED International Conference: Applied Informatics*, pp. 429-431, Innsbruck, Austria, February 1996.

[GAP91] Andrew H. Gee, Sreeram V. B. Aiyer, Richard W. Prager, "A Subspace Approach to Invariant Pattern Recognition using Hopfield Networks", Cambridge University Engineering Department Technical Report CUED/F-INFENG/TR62, Trumpington Street, CB2 1PZ, January 1991.
[ftp://svr-ftp.eng.cam.ac.uk/pub/pub/reports/gee_tr62.ps.Z]

[Gar92] Michael D. Garris, "A platform for Evolving Genetic Automata for Text Segmentation (GNATS)", *Science of Artificial Neural Networks*, Vol. 1710, pp. 714-724, SPIE, Orlando, Florida, April 1992.
[ftp://sequoyah.ncsl.nist.gov/pub/papers_preprints/gnats.ps.Z]

[Gar95] Michael D. Garris, "Evaluating Spatial Correspondence od Zones in Document Recognition Systems", *Proceedings of International Conference on Image Processong*, Washington D.C., October 1995.
[ftp://sequoyah.ncsl.nist.gov/pub/papers_preprints/ieee_icip95.ps.Z]

[GCB95] P. J. Grother, G. T. Candela, J. L. Blue, "Fast Implementation of Nearest Neighbor Classifiers", NIST Internal Report, 1995.
[ftp://sequoyah.ncsl.nist.gov/pub/papers_preprints/kdtree.ps.Z]

[GD96] Michael D. Garris, Darrin L. Dimmick, "Form Design for High Accuracy Optical Character Recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 18, No. 6, pp. 653-656, June 1996.

[GG95] Michael D. Garris, Patrick J. Grother, "Generalized Form Registration Using Structure-Based Techniques", *Proceedings of 5th Symposium on Document Analysis and Information Retrieval*, pp. 321-334, April 1996 and Technical Report NISTIR 5726, November 1995.
[ftp://sequoyah.ncsl.nist.gov/pub/nist_internal_reports/ir_5726.ps.Z]

[GJJ96] Earl Gose, Richard Johnsonbaugh, Steve Jost, *Pattern Recognition and Image Analysis*, Prentice Hall PTR, Upper Saddle River, NJ 07458, 1996.

[GK95] Srinivas Gazula, Mansur R. Kabuka, "Design of Supervised Classifiers Using Boolean Neural Networks", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 12, pp. 1239-1246, December 1995.

[Gos95] Bernard Gosselin, "Neural Networks Combination for Improved Handwritten Character Recognition", *Proceedings of IASTED International Conference: Signal and Image Processing – SIP-95*, Las Vegas, Nevada, USA, November 20-23, 1995.

[GP93] J. M. Gill, C. A. Peulevé, *Research Information Handbook of Assistive Technology for Visually Disabled Persons*, The Tiresias Consortium, 1993.

[GW92] Michael D. Garris, Charles L. Wilson, "A Neural Approach to Concurrent Character Segmentation and Recognition", In Southcon '92 Conference Record, pages 154-159, IEEE, Orlando, Florida, March 1992.
[ftp://sequoyah.ncsl.nist.gov/pub/papers_preprints/southcon-92.ps.Z]

[GW93] Philippe Gentric, Heini C.A.M. Withagen, "Constructive methods for a new classifier based on a Radial-Basis-Function neural network accelerated by tree", *New Trends in Neural Computation; Proceedings of IWANN'93*, pp. 125-130, Sitges, Spain, June 1993.

[GWB+92] M. D. Garris, C. L. Wilson, J. L. Blue, G. T. Candela, P. Grother, S. Janet, R. A. Wilkinson, "Massively Parallel Implementation of Character Recognition Systems", *Proceedings on Conference on Character Recognition and Digitizer Technologies*, Vol. 1661, pp. 269-280, SPIE, San Jose, California, February 1992., and Technical Report NISTIR 4750, January 1992.
[ftp://sequoyah.ncsl.nist.gov/pub/nist_internal_reports/ir_4750.ps.Z]

[GWW91] M. D. Garris, R. A. Wilkinson, C. L. Wilson, "Analysis of a Biologically Motivated Neural Network for Character Recognition," *Proceedings of the Analysis of Neural Network Applications*, ACM Press, George Mason University, May 1991.

[HK94] Basit Hussain, M. R. Kabuka, "A Novel Feature Recognition Neural Network and its Application to Character Recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 16, No. 1, pp. 98-106, January 1994.

[HM71] R. L. Hoffman, J.W. McCullough, “Segmentation methods for Recognition of Machine-Printed Cheracters”, *IBM Journal of Research and Development*, pp. 153-165, March 1971.

[Hon94] Vasant Honavar, “Toward Learning Systems That Use Multiple Strategies and Representations”, *Artificial Intelligence and Neural Networks: Steps Toward Principled Integration*, pp. 615-644, Honavar, V. and Uhr, L. (Ed.) New York: Academic Press, 1994.
["http://www.cs.iastate.edu/~honavar/Papers/learn94.ps]

[Hon95] Tao Hong, “*Degraded Text Recognition using Visual and Lingustic Context*”, PhD Dissertation, Faculty of the Graduated School of the State University of New York at Buffalo, September 1995.
[ftp://ftp.cs.buffalo.edu/pub/tech-reports/96-05.ps.Z]

[HS86] J. J. Hull, S. N. Srihari, “A Computaional Approach to Visual Word Recognition: Hypothesis Generation and Testing”, *Proceedings of Computer Vision and Pattern Recognition*, pp. 156-161, June 1986.

[HT96] Trevor Hastie, Robert Tibshirani, “Discriminant Adaptive Nearest Neighbor Classification”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 18, No. 6, pp. 607-615, June 1996.

[HYR86] A. Hashizume, P-S. Yeh, A. Rosenfield, “A Method of Detecting the Orientation of Aligned Components”, *Pattern Recognition Letters*, No. 4, pp. 125-132, 1986.

[II94] Mohan Inguva, Ashok Iyer, “Character Recognition using Neural Networks”, *Simulation Series*, Vol. 27, No. 2, pp. 159-162, Society for computer simulation, 1994.

[Jac88] Robert A. Jacobs, “Increased Rates of Convergence Through Learning Rate Adaptation”, *Neural Networks*, Vol. 1, pp. 295-307, 1988.

[JB92] Anil K. Jain, Sushil Bhattachatjee, “Text Segmentation Using Gabor Filters for Automatic Document Processing”, *Machine Vision and Applications*, No. 5, pp. 169-184, Springer-Verlag New York, 1992.

[Jel96] Jacek Jelonek, “ Generalization Capability of Homogeneous Voting Classifier Based on Partially Replicated Data”, *Proceedings of the National Conference on Artificial Intelligence*, (workshop on Integrating Multiple Learned Models), pp. 47-52, Portland, Oregon, August 1996.
[http://www.cs.fit.edu/~imlm/papers/jelonek.ps]

[JF93] T.T. Jervis, W.J. Fitzgerald, “Optimization Schemes for Neural Networks”, Cambridge University Engineering Department Technical Report CUED/F-INFENG/TR144, Trumpington Street, CB2 1PZ, August 1993.

[JKN+96] Dz-Mou Jung, M. S. Krishnamoorthy, George Nagy, Andrew Shapira, "N-Tuple Features for OCR Revisited", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 18, No. 7, pp. 734-745, July 1996.

[JKN93] F. Jenkins, J. Kanai, A. T. Nartker, "Using Ideal Images to Establish a Baseline of OCR Performance", 1993 Annual Research Report of Information Science Research Institute, University of Nevada, Las Vegas, Nevada, 1993. [http://www.isri.unlv.edu/info/tr/publications/AR-93.ps.gz]

[JS92] Ismail Jouny, Matthew Sheridan, "Character Recognition using a Multistage Neural Network", *Proceedings of SPIE* Automatic Object recognition II, Vol. 1700, Orlando, Florida, April 1992.

[KE96] Moshe Koppel, Sean P. Engelson, "Integrating Multiple Classifiers By Finding Their Areas of Expertise", *Proceedings of the National Conference on Artificial Intelligence*, (workshop on Integrating Multiple Learned Models), pp. 53-58, Portland, Oregon, August 1996. [http://www.cs.fit.edu/~imlm/papers/koppel.ps]

[KJ94] Cris Koutsougeras, Akhtar Jameel, "Experiments with Various Neural Architectures for Handwritten Character Recognition", *Midwest symposium on circuits and systems*, Vol. 37/V1, pp. 573-576, IEEE, 1994.

[Kon76] Блаже Конески, *Граматика на македонскиот литературен јазик*, Култура, Скопје, 1976.

[Kov68] V. A. Kovalevsky, *Character Readers and Pattern Recognition*, Spartan Books, Washington, D.C., 1968.

[KPB87] Simon Kahan, Theo Pavlidis, Henry S. Baird, "On the Recognition of Printed Characters of Any Font and Size", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 9, No. 2, pp. 274-287, March 1987.

[KRN+95] Junichi Kanai, Stephen V. Rice, Thomas A. Natker, George Nagy, "Automated Evaluation of OCR Zononig", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 1, pp. 86-90, January 1995.

[KS92] F. Kimura, M. Shridhar, "Segmentation-Recognition Algorithm for Zip Code Filed Recognition", *Machine Vision and Applications*, No. 5, pp. 199-210, Springer-Verlag New York, 1992.

[LCE89] Paul R. Lorczak, Alper K. Caglayan, Dave E. Eckhardt, "A Theoretical Investigation of Generalized Voters for Redundant Systems", *Proceedings of the Fault Tolerant Computing Systems*, Chicago, 1989.

[LDT95] David Lovell, Tom Downs, Ah Chung Tsoi, “Is the Neocognitron Capable of State-of-the-art Digit Recognition?”, Paper submitted to IEEE Transactions on Neural Networks, Cambridge University Engineering Department, Trumpington Street, CB2 1PZ, July 1995.
[ftp://svr-ftp.eng.cam.ac.uk/pub/pub/reports/lovell_neocognitron.ps.Z]

[Lee96] Seong-Whan Lee, “Off-Line Recognition of Totally Unconstrained Handwritten Numerals Using Multuilayer Cluster Neural Network”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 18, No. 6, pp. 648-652, June 1996.

[Lin91] Craig A. Lindley, *Practical Image Processing in C*, (in Esential Books on Graphics Programming, Dr. Dobb's CD-ROM Library, 1995), John Wiley & Sons, Inc., 1991.

[LLS92] Jiang Liu, Chung-Mong Lee, Ren-Ben Shu, “An Efficient Method for the Skew Normalization of a Document Image”, *International Conference on Pattern Recognition*, Vol. 11, No. 3, pp. 122-125, IEEE, 1992.

[LM89] E. Lecolinet, J-V. Moreau, “A New Sytem for Automatic Segmentation an Recognition of Unconstrained Zip Codes”, *Poceedings of the Sixth Scandinavian Conference on Image Analysis*, p. 585, Oulu, Finland, June 1989.

[LS95] Louisa Lam, Ching Y. Suen, “An Evaluation of Parallel Algorithms for Character Recognition”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 9, pp. 914-919, September 1995.

[LTS+92] T. Li, Y. Y. Tang, S. C. Suen, L. Y. Fang, A. J. Jennings, “A Structuraly Adaptive Neural Tree for the Recognition of Large Character Set”, *International Concerence on Pattern Recognition*, Vol. 11, No. 2, pp. 187-190, IEEE, 1992.

[Lu95] Yi Lu, “Machine Printed Character Segmentation – an Overview”, *Pattern Recognition*, Vol. 28, No. 1, pp. 67-80, 1995.

[Luc93] S. M. Lucas, “Optical Character Recognition with Hough Transformation Based Networks”, *IEE Colloquium Digest*, No. 106, pp. P7/1-5, Institution of Electrical Engineers, London, 1993.

[Man87] J. Mantas, “Methodologies in Pattern Recognition and Image Analysis – A Brief Survey”, *Pattern Recognition*, Vol. 20, No. 1, pp. 1-6, Pergamon Journals Ltd., 1987.

[Mat96] Ofer Matan, “On Voting Ensembles of Classifiers (extended abstract)”, *Proceedings of the National Conference on Artificial Intelligence*, (workshop on Integrating Multiple Learned Models), pp. 84-88, Portland, Oregon, August 1996.
[http://www.cs.fit.edu/~imlm/papers/matan.ps]

[MF95] M. Moreira, E. Fiesler, “Neural Networks with Adaptive Learning Rate and

Momentum Terms", IDIAP Technical Report 95-04, IDIAP, Martigny, Switzerland, October 1995.
[ftp://ftp.idiap.ch/pub/reports/1995/95-04.ps.Z]

[MGK93] Dragan Mihajlov, Dejan Ǵorǵeviḱ, Nevenka Kotevska, "Computer system for support of humans with damaged sight", *Zbornik na apstrakti ETAI 93*, Ohrid, 1993.

[MLL92] Eric Menu, Laurent Lefrere, Yves Lecourtier, "A New Method of Distorsion-Invariant Character Recognition", *International Conference on Pattern Recognition*, Vol. 11, No. 2, pp. 599-602, IEEE, 1992.

[MXM+95] Hong Ma, Dihua Xi, Xiaogang Mao, Yuan Y. Tang, Ching Y. Suen, "Document Analysis by Fractal Signature", *Document Analysis Systems* (L. Spitz & A. Dengel Edts.), World Scientific Series in Machine Perception & Artificial Intelligence, Vol. 14, 1995.

[Nes95] Martin Neschen, "Vector Quantisation Classifiers for Handwritten Character Recognition", Zentrum für Paralleles Rechnen (ZPR) Universität zu Köln Report No. 95-185, D-50931 Köln, Germany, also in *Proceedings of the ZEUS-95 Workshop*, Linkoping, Sweden, May 1995.
[ftp://ftp.zpr.uni-koeln.de/pub/paper/zpr95-185.ps.gz]

[Nes95a] Martin Neschen, " Hierarchical Binary Vector Quantisation Classifiers for Handwritten Character Recognition", Zentrum für Paralleles Rechnen (ZPR) Universität zu Köln Report No. 95-186, D-50931 Köln, Germany, also in *Proceedings of the 17th DAGM Symposion on Pattern Recognition*, Bielefeld, September 1995.
[ftp://ftp.zpr.uni-koeln.de/pub/paper/zpr95-186.ps.gz]

[NS93] Takehiro Nakayama, A. Lawrence Spitz, "European Language Determination from Image", *Proceedings of the International Conference on Document Analysis and Recognition*, Tsukuba Science City, Japan, October 1993.

[NSS+85] G. Nagy, S. Seth, S. D. Stoddard, "Document Analysis with an Expert System", *Proceedings, Pattern Recognition in Practice II*, Amsterdam, June 1985.

[NSV92] George Nagy, Sharad Seth, Mahesh Viswanathan, "A Prototype Document Image Analysis System for Technical Journals", *IEEE Computer Magazine*, Vol. 25, No. 7, pp. 10-22, July 1992.

[OG92] L. O'Gorman, "The Document Spectrum for Page Layout Analysis", *Proceedings IAPR Workshop on Structural and Sintactic Pattern Recognition*, World Scientific, pp. 270-279, Singapore, 1992.

[OK92] Lawrence O'Gorman, Rangachar Kasturi, "Document Image Analysis Systems", *IEEE Computer Magazine*, Vol. 25, No. 7, pp. 5-8, July 1992.

[Pae86] Alan W. Paeth, "A Fast Algorithm for General Raster Rotation", *Graphics Interface '86*, pp. 77-81, May 1986.

[Pav77] Theo Pavlidis, *Structural Pattern Recognition*, Springer-Verlag, 1997.

[Pav78] Theo Pavlidis, "A Review of Algorithms for Shape Analysis", *Computer Graphics and Image Processing*, Vol. 7, pp. 243-258, Academic Press, Inc., 1978.

[Pav84] Theo Pavlidis, *Algorithms for Graphics and Image Processing*, Computer Science Press, 1984.

[Pav93] Theo Pavlidis, "Recognition of Printed Text under Realistic Conditions", *Pattern Recognition Letters*, Vol. 14, No. 4, pp. 317-326, April 1993.

[Por95] Thomas E. Portegys, "A Search Technique for Pattern Recognition Using Relative Distances", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 9, pp. 910-912, September 1995.

[PSW95] R.K. Powalka, N. Sherkat, R.J. Whitrow, "Recognizer characterisation for combining handwriting recognition results at word level", Third *International Conference on Document Analysis and Recognition ICDAR'95*, Montreal, Canada, pp. 68-73, August 1995.
[http://www.doc.ntu.ac.uk/~inkdata/rkp/icdar95.2.ps]

[PSW95a] R.K. Powalka, N. Sherkat, R.J. Whitrow "Multiple recognizer combination topologies", *Proceedings of the Seventh Biennial Conference of the International Graphonomics Society*, pp. 128-129, London, Ontario, Canada, August 1995.
[http://www.doc.ntu.ac.uk/~inkdata/rkp/igs95.ps]

[PZ91] T. Pavlidis, J. Zhou, "Page Segmentation by White Streams", *Proceedings of Internacional Conference of Document Analysis and Recognition ICDAR'91*, Saint-Malo, France, September 1991.

[RBT+95] Richard Romero, Robert Berger, Robert Thibadeau, David Touretzky, "Neural Network Classifiers for Optical Chinese Character Recognition", Documentation of on-going project at Imaging Systems Lab, Robotics Institute, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, 1995.
[http://www.cs.cmu.edu/afs/cs.cmu.edu/project/www/papers/final.ps]

[RE94] Tony G. Rose, Lindsay J. Evett, "Handwriting Recognition using Semantic Information", in C. Faure, P. Keuss, G. Lorette & A. Vinter (Eds.), *Advances in Handwriting and Drawing - A Multi-disciplinary Approach*, EUROPIA Press, France, pp. 169-179, 1994.
[http://www.doc.ntu.ac.uk/Papers/tgr/paris_bo.ps]

[REJ94] G. Rose, L. J. Evett, A. C. Jobbins, "A Context Based Approach to Text Recognition", *Proc of 3rd Annual Symposium on Document Analysis and Information Retrieval*, pp. 219-227, Information Science Research Institutute, University of Nevada, Las Vegas, April 1994.
[http://www.doc.ntu.ac.uk/Papers/tgr/vegas_cr.ps]

[REL94] T. G. Rose, L. J. Evett, M. J. Lee, "Contextual Analysis for Text Recognition: A Comparison with Human Performance", Workshop Report, *AISB Quarterley 88*, pp. 41-44, Ed. Prof. Gerry Kelleher, 1994.
[http://www.doc.ntu.ac.uk/Papers/tgr/aisb_wk2.ps]

[RKN93] V. S. Rice, J. Kanai, A. T. Nartker, "An Evaluation of OCR Accuracy", 1993 Annual Research Report of Information Science Research Institute, University of Nevada, Las Vegas, Nevada, 1993.
[http://www.isri.unlv.edu/info/tr/publications/AR-93.ps.gz]

[RKN94] V. S. Rice, J. Kanai, A. T. Nartker, "The Third Annual Test of OCR Accuracy", 1994 Annual Research Report of Information Science Research Institute, University of Nevada, Las Vegas, Nevada, 1994.
[http://www.isri.unlv.edu/info/tr/publications/AR-94.ps.gz]

[RKN95] V. S. Rice, J. Kanai, A. T. Nartker, "The Fourth Annual Test of OCR Accuracy", 1995 Annual Research Report of Information Science Research Institute, University of Nevada, Las Vegas, Nevada, 1995.
[http://www.isri.unlv.edu/info/tr/publications/AR-95.ps.gz]

[Rob92] Tony Robinson, "Practical Network Design and Implementation", Presented at the Cambridge Neural Network Summer School, September 1992.
[ftp://svr-ftp.eng.cam.ac.uk/pub/pub/reports/robinson_cnnss92.ps.Z]

[RP93] J. Rocha, T. Pavlidis, "New Method for Word Recognition Without Segmentation", *Proceedings of SPIE Character Recognition Technologies*, Vol. 1906, pp. 74-80, 1993.

[RP93] Jairo Rocha, Theo Pavlidis, "New Method for Word Recognition Without Segmentation", *Proceedings of The Society of Photo-Optical Instrumentation Engineers*, Vol. 1906, pp. 74-80, 1993.

[RP94] Jairo Rocha, Theo Pavlidis, "A Shape Analysis model with Applications to a Character Recognition System", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 16, No. 4, pp. 393-404, April 1994.

[RP95] Jairo Rocha, Theo Pavlidis, "Character Recognition Without Segmentation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 9, pp. 903-909, September 1995.

[RSZ+94] Jairo Rocha, Bill Sakoda, Jiangyin Zhou, Theo Pavlidis, "Deffered Interpretation of Grayscale Saddle Features for Recognition of Touching and Broken Characters", *Proceedings of the Society of Photo-Optical Instrumentation Engineers*, No. 2181, pp. 342, 1994.

[RTT96] Richard Romero, David Touretzky, Robert Thibadeau, "Optical Chinese Character Recognition using Probabilistic Neural Networks", Documentation of on-going project at Imaging Systems Lab, Robotics Institute, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, 1995.
[http://www.cs.cmu.edu/afs/cs.cmu.edu/project/pcvision/www/papers/chinese-pnn/chinese-pnn.frame.ps]

[SCD94] Patrice Y. Simard, Yann Le Cun, John S. Denker, "Memory-Based Character Reognition Using a Transformation Invariant Metric", *Proceedings of the 12th IAPR International Conference on Pattern Recognition*, Volume II, pp. 262-267, Jerusalem, Israel, October 1994.

[Sch89] Robert J. Schalkoff, *Digital Image Processing and Computer Vision*, John Wiley & Sons, Inc., 1989.

[SDH+95] Steven Salzberg, Arthur L. Delcher, David Heath, Simon Kasif , "Best-Case Results for Nearest-Neighbor Learning", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 6, pp. 599-608, June 1995.

[Sen92] Andrew W. Senior, "Off-line Handwriting Recognition: A Review and Experiments", Cambridge University Engineering Department Technical Report CUED/F-INFENG/TR105, December 1992.
[ftp://svr-ftp.eng.cam.ac.uk/pub/pub/reports/senior_tr105.ps.Z]

[Sen94] Andrew W. Senior, "*Off-line Cursive Handwriting Recognition using Recurrent Neural Networks*", PhD Disertation, Cambridge University Engineering Department, Trumpington Street, CB2 1PZ, September 1992.
[ftp://svr-ftp.eng.cam.ac.uk/pub/pub/reports/senior_thesis.ps.gz]

[SF92] Andrew W. Senior, F. Fallside, "Off-line Handwriting Recognition by Recurrent Error Propagation Networks", *Proceedings of British Machine Vision Conference*, York, September 1992.
[ftp://svr-ftp.eng.cam.ac.uk/pub/pub/reports/senior_bmvc.ps.Z]

[SH+92] Shigeyoshi Shimotsuji, Osamu Hori, Meiko Asano, Kaoru Suzuki, Fumihiko Hoshino, Toshiaki Ishii, "A Robust Recognition System for a Drawing Superimposed on a Map", *IEEE Computer Magazine*, Vol. 25, No. 7, pp. 56-59, July 1992.

[Sha95] Andrew Shapira, "Experiments on Generation of Distinguishing N-Tuples for Selected Character Dichotomies", Technical Report No. ECSE-OCR-18DEC95, ECSE Department, Rensselaer Polytehnic Institute, Troy, NY 12180, December 1995.
[http://www.cs.rpi.edu/~shapiraa/papers/genexpts/t.ps.gz]

[Shu94] Alexander Shustorovich, "A Subspace Projection Approach to Feature Extraction: The Two-Dimensional Gabor Transform for Character Recognition", *Neural Networks*, Vol. 7, No. 8, pp. 1295–1301, Pergamon Journals Ltd., 1994.

[SL+92] Sargur N. Srihari, Stephen Lam, V. Govindaraju, Rohini Srihari, Jonathan Hull, "Document Understanding: Research Directions", Technical Report CEDAR-TR-92-1, State University of New York at Buffalo, Amherst, NY 14228-2567, May 1992.
[http://www.cedar.buffalo.edu/Publications/TechReps/Postscript/Survey.ps]

[SL94] Sargur N. Srihari, Stephen W. Lam, "Character recognition", Technical Report CEDAR-TR-95-1, State University of New York at Buffalo, Amherst, NY 14228-2567, October 1995.
[http://www.cedar.buffalo.edu/Publications/TechReps/Postscript/OCR.ps]

[SMK+94] Stephen J. Smith, Mario O. Bourgoin, Karl Sims, Harry L. Voorheees, "Handwritten Character Classification Using Nearest Neighbor in Large Databases", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 16, No. 9, pp. 915-919, September 1994.

[SP92] Takashi Saitoh, Theo Pavlidis, "Page Segmentation without Rectangle Assumption", *Proceedings of Internacional Conference on Pattern Recognition*, pp. 277-280, IEEE, 1992.

[SP95a] Jaakko Sauvola, Matti Pietikäinen, "Skew Angle Detection using Texture Direction Analysis", *9th Scandinavian Conference on Image Analysis*, pp. 1099-1106, June 6-9, UPPSALA, Sweden, 1995.
[http:\\ee.oulu.fi\~jjs\]

[SP95b] Jaakko Sauvola, Matti Pietikäinen "Page Segmentation and Classification Using Fast Feature Extraction and ConnectivityAnalysis", *Third International Conference on Document Analysis and Recognition (ICDAR'95)*, pp. 1127-1131, Montreal, Canada, August 14-16, 1995.
[http://ee.oulu.fi/~jjs]

[Spe88] D. F. Specht, "Probabilistic Neural Networks for Classification, Mapping or Associative Memory", *Proceedings of the International Conference on Neural Networks*, 1988.

[Spi94] A. Lawrence Spitz, "Text Characterization by Connected Component Transformations", *Proceedings of SPIE Symposium on Electronic Imaging Science & Technology*, San Jose, CA, pp. 97–105, February 1994.

[Spi95a] A. Lawrence Spitz, “An OCR Based on Character Shape Codes and Lexical Information”, *Proceedings of the Third International Conference on Document Analysis and Recognition*, pp. 723-728, Montreal, Canada, 1955.

[Spi95b] A. Lawrence Spitz, “Using Character Shape Codes for Word Spotting in Document Images”, *Shape, Structure and Recognition*, D. Dori and A. Bruckstein, eds., World Scientific, Singapore, 1995.

[SR95] Andrew Senior, Tony Robinson, “Forward-backward Retraining of Recurrent Neural Networks”, *Proceedings of NIPS '95*, November 1995.
[ftp://svr-ftp.eng.cam.ac.uk/pub/pub/reports/senior_fbrnn.ps.gz]

[Sri96] Sargur N. Srihari, “Recent advances in off-line handwriting recognition at CEDAR”, Technical Report CEDAR-TR-96-1, Center of Excellence for Document Analysis and Recognition and Department of Computer Science, State University of New Yourk at Buffalo, Buffalo, NY 14260, September 1996.
[http://www.cedar.buffalo.edu/Publications/TechReps/OLHWR/offlinehwr.html]

[SS94] Penelope Sibun, A. Lawrence Spitz, “Language determination: Natural Language Processing from Scanned Document Images”, *Proceedings of the 4th Conference on Applied Natural Language Processing*, pp. 15--21, Stuttgart, Germany, 1994.

[SS94] Sargur N. Srihari, Rohini K. Srihari, “Written Language Input”, Technical Report CEDAR-TR-94-1, State University of New York at Buffalo, Amherst, NY 14228-2567, May 1994.
[http://www.cedar.buffalo.edu/Publications/TechReps/Postscript/WLR.ps]

[SS95] Sargur N. Srihari, Rohini K. Srihari, Chapter 2: “Written Language Input”, *Human Language Technologies*, pp. 71-108, 1995.
[http://www.cse.ogi.edu/CSLU/HLTsurvey/]

[SSR94] S. N. Srihari, Y. C. Shin, V. Ramanaprasada, D. S. Lee, “A System to Read Names and Addresses on Tax Forms”, Technical Report CEDAR-TR-94-2, Center of Excellence for Document Analysis and Recognition and Department of Computer Science, State University of New Yourk at Buffalo, Buffalo, NY 14260, October 1994.
[http://www.cedar.buffalo.edu/Publications/TechReps/NABR.ps]

[SYT94] Takashi Saitoh, Toshifumi Yamaai, Michiyoshi Tachikawa, “Document Image Segementation and Layout Analysis”, *IEICE Transactions on Information and System*, Vol. 77, No. 7, pp. 778, 1994.

[SZP93] Bill Sakoda, Jiangying Zhou, Theo Pavlidis, “Address Recognition System Based on Feature Extraction from Gray Scale”, *Proceedings of the Society of Photo-Optical Instrumentation Engineers*, No. 1906, pp. 21, 1993.

[TA92] S. Tsujimoto, H. Asada, "Major Components of a Complete Text Reading System", *Proceedings of IEEE*, Vol 80, No. 7, pp. 1133-1149, July 1992.

[TBB+93] Kazem Taghava, Julie Borsack, Bryan Bullard, Allen Condit, "Post-Editing through Approximation and Global Correction", 1993 Annual Research Report of Information Science Research Institute, University of Nevada, Las Vegas, Nevada, 1993.
[http://www.isri.unlv.edu/info/tr/publications/AR-93.ps.gz]

[TBC+93] Kazem Taghava, Julie Borsack, Allen Condit, Srinivas Erva, "The Effects of Noisy Data on Text Retrieval", 1993 Annual Research Report of Information Science Research Institute, University of Nevada, Las Vegas, Nevada, 1993.
[http://www.isri.unlv.edu/info/tr/publications/AR-93.ps.gz]

[TBC+94] Kazem Taghava, Julie Borsack, Allen Condit, Jeff Gilberth, "Results and Implications of the Noisy Data Projects", 1994 Annual Research Report of Information Science Research Institute, University of Nevada, Las Vegas, Nevada, 1994.
[http://www.isri.unlv.edu/info/tr/publications/AR-94.ps.gz]

[TBC94] Kazem Taghava, Julie Borsack, Allen Condit, "An Expert System for Automatically Correcting OCR Output", *Proceedings of IS&T/SPIE Symposium on Electronic Imaging Science and Technology*, San Jose, CA, January 1994.
[http://www.isri.unlv.edu/info/tr/publications/Taghva94a.ps.gz]

[TCB+94] Kazem Taghava, Allen Condit, Julie Borsack, Srinivas Erva, "Structural Markup of OCR Generated Text", University of Nevada Tehnical Report 94-02, Las Vegas, Nevada, March 1994.
[http://www.isri.unlv.edu/info/tr/publications/Taghva94-02.ps.gz]

[TCB95] Kazem Taghava, Allen Condit, Julie Borsack, "An Evaluation of an Automatic Markup System", IS&T/SPIE Symposium on Electronic Imaging Science and Technology, San Jose, CA, February, 1995.
[http://www.isri.unlv.edu/info/tr/publications/Taghva95a.ps.gz]

[TF95] Georg Thimm, Emile Fiesler, "Evaluating Pruning Methods", *Proceedings of International Symposium on Artificial Neural Networks (ISANN'95)*, 1995.
[ftp://ftp.idiap.ch/pub/papers/neural/thimm.pruning-hop.ps.Z]

[TF96] Georg Thimm, Emile Fiesler, "Neural Network Pruning and Pruning Parameters", *The 1st Workshop on Soft Computing*, 1996.
[ftp://ftp.idiap.ch/pub/papers/neural/thimm.prune-params.ps.Z]

[TF97] G. Thimm, E. Fiesler, "High Order and Multilayer Perceptron Initialization", Accepted for publication in IEEE Transactions on Neural Networks, 1997.
[ftp://ftp.idiap.ch/pub/papers/neural/thimm.weight-initi.ps.Z]

[TG95] Kagan Tumer, Joydeep Ghosh, “Order Statistics Combiners for Neural Classifiers”, *Proceedings of the World Congress on Neural Networks*, pp. I:31-34, Washington, DC, July 1995.
[ftp://ftp.lans.ece.utexas.edu/pub/papers/order_stat_wcnn95.ps.Z]

[TG96] Kagan Tumer, Joydeep Ghosh, “Classifier Combining: Analytical Results and Implications”, *Proceedings of the National Conference on Artificial Intelligence*, (workshop on Integrating Multiple Learned Models), pp. 126-132, Portland, Oregon, August 1996.
[http://www.cs.fit.edu/~imlm/papers/tumer.ps]

[TG96a] Kagan Tumer, Joydeep Ghosh, “Error Correlation and Error Reduction in Ensemble Classifiers”, *Connection Science, Special issue on combining artificial neural networks: ensemble approaches*, Vol. 8, No. 3 & 4, pp. 385-404, December 1996.
[ftp://ftp.lans.ece.utexas.edu/pub/papers/error_corr_cs.ps.Z]

[TIFF6] Aldus & Microsoft, “TIFF Revision 6.0”, June 1992.
[http://www.adobe.com/supportservice/devrelations/PDFS/TN/TIFF6.pdf]
[ftp://ftp.sgi.com/graphics/tiff/TIFF6.ps.Z]

[TJ95] Øivind Due Trier, Anil K. Jain, “Goal-Directed Evaluation of Binarization Methods”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 12, pp. 1191-1201, December 1995.

[TK88] Atsushi Tanaka, Masatoshi Kameyama, “Image Rotating System By an Arbitrary Angle”, U.S. Patent 4,759,076, Mitsubishi Denki Kabushiki Kaisha, July 19, 1988.

[TR93] E. Tuv, A. N. Refenes, “Removal of Catastrophic Noise in Hetero-Associative Training Samples”, *Microprocessing and Microprogramming*, Vol. 38, pp. 697-704, North-Holland, 1993.

[TS92] Yuan Y. Tang, Ching Y. Suen, “Parallel Character Recognition Based on Regional Projection Transformation (RPT)”, *International Conference on Pattern Recognition*, Vol. 11, No. 2, pp. 631-634, IEEE, 1992.

[TSW90] Charles C. Tappert, Ching Y. Suen, Toru Wakahara, “The State of the Art in On-Line Handwriting Recognition”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, No. 8, pp. 787-808, August 1990.

[TT95] Øivind Due Trier, Torfinn Taxt, “Evaluation of Binarization Methods for Document Images”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 3, pp. 312-315, March 1995.

[Tum96] Kagan Tumer, “*Linear and Order Statistics Combiners for Reliable Pettern Classification*”, PhD Dissertation, The University of Texas at Austin, May 1996.

[VDK+70] *Правопис на македонскиот литературен јазик со правописен*

речник, редакциски одбор: Божо Видоески, Тодор Димитровски, Кирил Конески, Крум Тошев (претседател), Рада Угринова-Соколовска, Графички завод "Гоце Делчев", Скопје, 1970.

[VT92] Pascal Vaxivière, Karl Tombre, "Celesstin: CAD Conversion of Mehanical Drawings", *IEEE Computer Magazine*, Vol. 25, No. 7, pp. 46-54, July 1992.

[Waa95] W.P. de Waard, "An Optimised Distance Method for Character Recognition", *Pattern Recognition Letters*, Vol. 16, No. 5, pp. 499-506, May 1995.

[WCW82] F. M. Wahl, K. Y. Wong, R. G. Casey, "Block Segmentation and Text Extraction in Mixed Text/Image Documents", *Computer Graphics and Image Processing*, Vol. 20, pp. 375-390, 1982.

[Wei80] Carl F.R. Weiman, "Continuous Anti-Aliased Rotation and Zoom of Raster Images", Computer Graphics, (*SIGGRAPH '80 Proceedings*), Vol. 14, No. 3, pp. 286-293, July 1980.

[WG92] R. Allen Wilkinson, Michael D. Garris, "Comparison of Massively Parallel Hand-print Segmenters", In D. P. Casasent, editor, *Intelligent Robots and Computer Vision: Algorithms, Techniques, and Active Vision*, Vol. 1825, SPIE, Boston, Massachusetts, 1992 and Technical Report NISTIR 4923, September 1992.
[ftp://sequoyah.ncsl.nist.gov/pub/nist_internal_reports/ir_4923.ps.Z]

[WGJ+92] R. Allen Wilkinson, Jon Geist, Stanley Janet, Patrick J. Grother, Christophrt J. C. Burges, Robert Creecy, Bob Hammond, Jonathan J. Hull, Norman W. Larsen, Thomas P. Vogl, Charles L. Wilson, "The First Census Optical Character Recognition Systems Conference", National Institute of Standards and Technology, Technical Report NISTIR 4912, August 1992.
[ftp://sequoyah.ncsl.nist.gov/pub/nist_internal_reports/ir_4912.ps.Z]

[Wil91] R. Allen Wilkinson, "Segmenting Text Images With Massively Parallel Machines", In D. P. Casasent, editor, *Intelligent Robots and Computer Vision*, Vol. 1607, pp. 312-323, SPIE, Boston, Massachusetts, November 1991.
[ftp://sequoyah.ncsl.nist.gov/pub/papers_preprints/boston_91.ps.Z]

[Wil92] C. L. Wilson. "Effectiveness of Feature and Classifier Algorithms in Character Recognition Systems," Technical Report NISTIR 4995, December 1992 and In D. P. D'Amato, editor, Vol. 1906 SPIE, San Jose, CA, 1993.
[ftp://sequoyah.ncsl.nist.gov/pub/nist_internal_reports/ir_4995.ps.Z]

[Wol90] George Wolberg, *Digital Image Warping*, (in Esential Books on Graphics Programming, Dr. Dobb's CD-ROM Library, 1995), IEEE, 1990.

[WS89] Dacheng Wang, Sargur N. Srihari, "Classification of Newspaper Image Blocks Using Texture Analysis", *Computer Vision, Graphics and Image Processing*, Vol. 47, pp. 327-352, Academic Press Inc., 1989.

[XKS92] Lei Xu, Adam Krzyżak, Ching Y. Suen, "Methods of Combining Multiple

Classifiers and Their Applications to Handwriting Recognition", *IEEE Transactions on Sytems, Man, and Cybernetics*, Vol. 22, No. 3, May 1992.

[Yan93] Hong Yan, "Skew Correction of Document Images Using Interline Cross-Corelation", *CVGIP: Graphical Models and Image Processing,* Vol. 55, No. 6, pp. 538-543, Academic Press Inc., November 1993.

[ZP93] Jiangying Zhou, Theo Pavlidis, "Disambiguation of Characters by a Second-Stage Classifier", Proceedings of the Society of Photo-Optical Instrumentation Engineers (SPIE), Vol. 1906, pp. 166-171, 1993.

[ZP94] Jiangying Zhou, Theo Pavlidis, "Discrimination of Characters by a Multi-Stage Recognition Process", *Pattern Recognition*, Vol. 27, No. 11, pp. 1539-1549, 1994.