БЪЛГАРСКА АКАДЕМИЯ НА НАУКИТЕ
ИНСТИТУТ ПО ИНФОРМАЦИОННИ И КОМУНИКАЦИОННИ ТЕХНОЛОГИИ

Десислава Николова Бояджиева

КОМБИНИРАН ПОДХОД ЗА РАЗПОЗНАВАНЕ НА ОН-ЛАЙН ПОДПИСИ

ДИСЕРТАЦИЯ

за придобиване на образователната и научна степен „доктор“

по специалност 01.01.12. "Информатика"

в професионално направление 4.6. "Информатика и компютърни науки"

Научен ръководител: доц. д-р Георги Глухчев

София, 2014 г.

Благодарности

Изказвам своята сърдечна благодарност на научния си ръководител доц. д-р Георги Глухчев за компетентните напътствия, професионалните съвети и търпението, оказани при разработването на дисертационния труд.

Благодаря и на колегите от колектива на секция ”Обработка на изображения и разпознаване на образи” с ръководител доц. д-р Димо Димов за предоставянето на образци от подписите им, необходими за провеждане на част от експериментите.

Благодаря на съпруга и близките ми за подкрепата и разбирането, които проявиха по време на моята работа.

Работата по дисертационния труд е осъществена с подкрепата на Проект: BG051PO001-3.3.04/40 по Оперативна програма “Развитие на човешки ресурси” на ЕСФ и МОМН - Република България.

Съдържание

Списък на фигурите

Списък на таблиците

Речник на термините

Термин (бълг.)	**Термин (англ.)**	**Съкр.**
ROC крива	Receiver Operating Characteristic	ROC
Активационна функция	Transfer Function	
Алгоритъм за спускане по градиента	Gradient Descent Algorithm	
Апроксимация на данни	Data Fitting	
В реално време	Online	
Валидираща извадка	Validation set	
Вероятностни невронни мрежи	Probabilistic Neural Networks	PNN
Времеви отпечатък	Timestamp	
Времеви последователности от данни	Time Data Series	
Гаусов смесен модел	Gaussian Mixture Model	GMM
Грешка при обобщаване	Generalization Error	
Грешка на класификация	Classification Error	PCTError
N-кратна крос валидация	N-Fold Cross-Validation	N-Fold CV
Leave one out крос валидация	Leave one out Cross-Validation	LOOCV
Динамично времево изкривяване	Dynamic Time Warping	DTW
Доверителен интервал	Confidence Interval	
Дял на лъжливо отрицателните образци	False Reject Rate	FRR
Дял на лъжливо положителните образци	False Accept Rate	FAR
Дял на истински отрицателните образци	True Reject Rate	TRR, Specifity
Дял на истински положителните образци	True Accept Rate	TAR, Sensitivity
Идентификация, установяване на самоличност	Identification	
Изкуствени невронни мрежи	Artificial Neural Networks	НМ
Изместена оценка	Biased estimate	
Инструментариум	Toolbox	
Качество	Performance	
Класификатор с Парзенови прозорци	Parzen Windows Classification	

Личен цифров помощник	Personal Digital Assistant	PDA
Лъжливо отрицателен образец	False Negative	FN
Лъжливо положителен образец	False Positive	FP
Истински отрицателен образец	True Negative	TN
Истински положителен образец	True Positive	TP
Мажоритарно гласуване	Majority Voting	
Матрица на класификация	Confusion Matrix	
Метод на опорните вектори	Support Vector Machines	SVM
Метод на *k*-най-близки съседи	*k*-Nearest Neighbors Algorithm	*k*NN
Многослоен перцептрон	Multilayer Perceptron	MLP
Модалност, биометрична характеристика	Trait, Modality	
Мрежови признаци	Grid Features	
Мултикласификатор (комбинация от класификатори)	Multiple Classifier Systems	MCS
Мултимодални биометрични системи	Multimodal Biometric Systems	
Невронна мрежа с обратно разпространение на грешката	Feedforward Backpropagation Neural Networks	
Невронни мрежи с право разпространение на сигнала, прави мрежи	Feedforward Neural Networks	FNN
Невронни мрежи с радиални базисни функции	Radial Basis Functions Neural Networks	RBF NN
Недостатъчно настройване на модел	Underfitting	
Прости фалшифицирани подписи	Simple Forgery Signatures	
Обобщена мрежа	Generalized Net	GN
Области с нулев натиск	Zero pressure regions	
Обработващи елементи, неврони	Processing Units	
Образец, пример	Sample	
Образи	Patterns	
Обучаваща извадка	Training set	
Остатъчна сума от квадрати	Residual Sum of Squares	RSS
Оригинален, собствен подпис	True Signature	
Отклонение, изместване	Bias	
Установяване на самоличност	Authentication	
r-подмножество от променливи, които напускат регресионния модел		*r*-subset
p-подмножество от променливи, които		*p*-subset

остават в регресионния модел		
Пакет за разработка на приложения	Software Development Kit	SDK
Подбор, избор на класификатори	Classifier Selection	
Потребител, участник	User	
Почерк, ръкописен текст	Handwriting	
Прецизност на класификация	Classification Precision	Precision
Праг	Threshold	
Пренастройване	Overfitting	
Признаци	Features	
Признаци, силно отдалечени от групата данни	Outliers	
Протокол за събиране на данни	Acquisition Protocol	
Първоначално влизане в система, включване	Login	
Равен процент на грешките	Equal Error Rate	EER
Редукции на m-подмножества	m-variable reductions	
Редукция на регресионната сума от квадрати, получена от премахването на подмножество от r променливи	Reduction in the regression sum of squares	$\mathrm{Re}\,d_r$
Редукция от премахването на променлива	Univariate reduction	
Система за разпознаване по подпис	Signature Recognition System	СРП
Скорост на обучение	Learning Rate	
Скорост на предаване на данните	Sampling Rate	
Скрити Марковски модели	Hidden Markov Models	HMM
Неумели фалшифицирани подписи	Random Forgery Signatures	
Средно квадратична грешка	Mean square error	mse
Стопиращо условие	Stopping Condition	
Тестова извадка	Test set	
Точност на класификация	Classification Accuracy	AC
Умели фалшифицирани подписи	Skilled Forgery Signatures	
Усредняване	Averaging	
Фалшификатор, имитатор	Impostor	
Фалшифициран (подправен) подпис	Forgery Signature	
Сливане на класификатори	Classifier Fusion	
Функция за оценка на качеството	Performance Function	
Цифрово мастило	Digital ink	
Щрих	Stroke	

Увод

Верификацията и идентификацията по подпис, обединени от общия термин разпознаване по подпис, са задачи от областта на разпознаването на образи и анализа на документи. Целта на верификацията е потвърждаването или отхвърлянето на заявена самоличност, а на идентификацията - установяването на самоличността на даден потребител. В зависимост от начина на събиране на подписи, системите за разпознаване по подпис са статични (оф-лайн) и динамични (он-лайн). При първите подписът се привежда в цифров вид чрез сканиране или заснемане с фотокамера след като вече е бил положен върху хартия и по-нататък се работи с неговото полутоново статично изображение, докато при вторите подписите се полагат директно чрез дигитална писалка върху устройства като графичен таблет или друг вид сензорен екран, като по този начин автоматично се получава едновременно и статична, и динамична информация за подписа.

Тъй като подписът е често употребявано, удобно, сигурно и широко възприето средство за установяване на самоличност, с неинвазивно, бързо, лесно и естествено за потребителите въвеждане в биометричните системи, то през последните години се наблюдава засилен интерес към разработките в областта. Разработват се нови методи и алгоритми, част от които се внедряват в практически приложения [14, 17, 27, 51, 101, 102], като се наблюдава превес на он-лайн пред оф-лайн методите за разпознаване по подпис. С изобретяването на все повече и повече нови устройства с дигитални писалки и сензорни екрани, които предоставят възможност още с полагане на подписа да се разполага с неговия дигитален формат, се откриват нови хоризонти пред областите на приложение на он-лайн системи за разпознаване по подпис. Така практически всяко приложение за достъп до система, ресурс или устройство , за което се изисква парола за достъп, може да се замени с он-лайн верификация по подпис, а он-лайн идентификацията по подпис успешно да се прилага в случаи на хартиен носител. Изобщо, широките области на приложение в създаването и внедряването на системи за установяване на самоличност определят актуалността на задачата в световен мащаб, а оттам и актуалността на дисертационната работа.

Глава 1. Разпознаване на подписи

1.1 Постановка на задачата за разпознаване на подписи

1.1.1 Верификация и идентификация на самоличност

Верификацията и идентификацията на самоличност, обединени от общия термин разпознаване на самоличност, са задачи от областта на разпознаването на образи. Целта на верификацията на самоличност е потвърждаването или отхвърлянето на заявена самоличност (“Аз ли съм този, за който се представям?”) чрез извършването на 1:1 сравняване на някои специфични признаци, докато при идентификацията, също позната като 1 : *N* сравнение, се цели установяването на самоличността (“Кой съм аз?”) на даден потребител чрез сравняване на някои специфични признаци с тези на *N* участници от базата данни. Верификацията би могла да се интерпретира като класификация в два класа “аз съм – аз не съм”, идентификацията като отнасяне към някой от N класа „аз съм еди-кой си“. В Таблица 1.1 са отразени разликите между двата типа разпознаване на самоличност.

Таблица 1.1 Разлика между системите за верификация и идентификация

Тип разпознаване	**Верификация**	**Идентификация**
Клас	1 клас	N класа
Вид сравнение	1:1	1:N
Изход	Приемане или отхвърляне	Избор на клас (Идентификатор)

Системите за верификация в общия случай са по-бързи и по-точни от тези за идентификация, които от своя страна изискват повече изчислителни ресурси, поради необходимостта от извършването на по-голям брой сравнения.

В областта на верификацията и идентификацията на самоличност се провеждат активни изследвания и разработки. Съществуващите методи за верификация на самоличност, използвани в съвременните системи за установяване на самоличност биха могли да се групират в следните типове [23]: базирани на притежание, базирани на знание и биометрични. Те са представени в Таблица 1.2. При верификацията, базирана на знания, се разчита на това потребителите да запаметят тайна информация, която в последствие да бъде използвана за потвърждаване на самоличност пред системата. Обикновено това е парола, ПИН код и др. При верификацията, базирана на притежание се изисква от потребителите физически да притежават даден предмет – магнитни карти, смарт карти, USB донгъли и др. Основните недостатъци на първите два метода са: (1) знанието на дадена информация би могло да се забрави; (2) предметите биха могли

да се загубят, подправят или лесно да се дублират; (3) и знанието, и предметите биха могли да се споделят или откраднат. Това, разбира се, е недопустимо в приложения, които изискват високо ниво на сигурност, като достъп до банкова сметка, използване на кредитна карта или физически достъп до ресурс.

Таблица 1.2 Типове методи за установяване на самоличност

Тип	**Пример**	**Недостатъци**
Базирани на знание	ID, парола, PIN	Паролите биха могли да се отгатнат или забравят
Базирани на притежание	Карти, баджове, ключове	Предметите могли да се изгубят, откраднат или да им се извади дубликат
Биометрични	Пръстови отпечатъци, лице, ирис, глас, подпис, поведение	Събирането на биометрични данни е скъпо и не много удобно за индивида

1.1.2 Биометрични данни и подписът като биометрични данни

Необходимото ниво на сигурност би могло да се предложи от системи за разпознаване, базирани на биометрични данни. Биометриката е наука, която се базира на измерванията на различни физически, физиологически и поведенчески характеристики. Терминът произлиза от гръцките думи био (живот) и метрика (измерване). Биометричните данни предоставят възможност за удостоверяване на самоличност и за разлика от другите възможни средства са уникални [24]. Някои от характеристиките са неизменни за всеки индивид през целия му живот (ирис, пръстови цвят на кожата.), други се формират в процеса на развитието му и могат да търпят промени в различни периоди от живота (подпис, глас, лице, походка). Те се отнасят към поведенческите характеристики.

Първите се основават на директно измерване на елементи от човешкото тяло, докато вторите използват данни, получени при извършване на дадено действие и следователно са свързани индиректно с физиката на човека. В Таблица 1.3 са представени двата типа биометрични данни.

Таблица 1.3 Типове биометрични данни

Тип биометрични данни	**Примери**
Физически	Пръстови отпечатъци, ретина, лице, ирис, геометрия на длан та, вени
Поведенчески	Глас, почерк, походка, подпис

През последните години са формулирани изисквания към биометричните системи и биометричните модалности, обобщени в [11], а именно биометричните системи да бъдат точни, надеждни, на приемлива цена, бързи, лесно възприемани от потребителите, а биометричните модалности да бъдат универсални, уникални, постоянни и събираеми. За съжаление не съществува биометрична модалност, която напълно да отговаря на тези изисквания. При проектирането на биометрична система се избират една или повече биометрични модалности в зависимост от приложението й. Биометричните системи, проектирани на базата на повече от една биометрична характеристика, носят наименованието мултимодални системи [14]. Използването на мултимодална система повишава сигурността на разпознаването. В една такава система би могло да се предостави възможност за избор измежду различни модалности като по този начин тя става достъпна за по-голям кръг от потребители, включително и за хора с увреждания.

Според броя на използваните биометрични модалности, сензори и признакови множества, биометричните системи работят при един от следните сценарии [74]:

1. *Една модалност, няколко сензора*

   Чрез различни по вид сензори се получава информация за една единствена биометрична модалност. Получената информация би могло да се комбинира. Например за събиране на подписи сензорите биха могли да бъдат графичен таблет, графичен таблет с хартия за полагане на подписа, телевизионна камера, личен персонален помощник (PDA) и др.;

2. *Повече от една биометрична модалност*

   За даден потребител би могло да се комбинира информацията, постъпваща от различни сензори за всяка от различните биометрични модалности;

3. *Няколко единици от една и съща модалност*

   Това е не скъп начин за подобряване на качеството на биометричната система, тъй като не се изисква наличието на допълнителни сензори. В този случай може да се интегрира информацията от няколко единици на една и съща модалност (например ирисите и на двете очи [77]) за даден потребител;

4. *Една модалност, няколко класификатора*

   Това е най-ефективният и евтин начин за подобряване на качеството на биометричните системи. Входните данни за единствена биометрична характеристика се получават от единствен сензор и по нататък се обработват от

няколко класификатора като всеки от тях работи върху общо множество от признаци или използва индивидуално.

Подписът се отнася към една от най-често използваните модалности за установяване на самоличност. Една от основните причини за това е многолетната практика за удостоверяване на самоличност чрез подписване. По тази причина той се възприема като нещо естествено и не предизвиква възражение. Заедно с това полагането му не е свързано с неудобство за лицето поради неинвазивното му бързо и лесно реализиране.

Минималните изисквания към размера на базата данни от образци на подписи, независимостта на подписа от националността на потребителите и факта, че всеки човек би могъл да промени подписа си по свое желание (за разлика от други биометрични модалности) са сред допълнителните предимства на подписа. Друга причина, възникваща напоследък, е свързана с новата генерация лични персонални помощници поради това, че голяма част от тях приемат за вход ръкописен текст.

Заедно с тези предимства подписът притежава и сериозен недостатък, произтичащ от промяната му през живота на човека. Обикновено почеркът се формира до към 25-тата година, запазвайки устойчивост до към 60-тата година. Промени в динамиката и графиката на подписа могат да се предизвикат от различни видове заболявания, най-често неврологични, различни лекарства и опиати, стресови ситуации и др. Към недостатъците трябва да се добави и възможността за сравнително лесно имитиране на графиката на подписа. Тези неща, заедно с естествената изменчивост, правят задачата за разпознаването му трудна [11]. Това изисква внимателен подход към избора на признаците за представяне на подписа и правилата за класификация, отчитайки различната тежест на двете възможни грешки: лъжливо-положителен резултат, когато подправен подпис е приет за собствен и лъжливо-отрицателен, когато собствен подпис е класифициран като подправен. Ясно е, че първата грешка е много по-сериозна, докато при втората лицето обикновено има възможност да се подпише отново.

1.1.3 Системи за разпознаване по подпис (СРП)

Разпознаването по подпис представлява процеса на потвърждаване на самоличността на потребител въз основа на неговия подпис [17] и е обект на изследване в областта на разпознаването на образи и анализа на документи. В системите за разпознаване по подпис се фокусира върху извличането на специфични характеристики от образците на подписите за всеки потребител, които го разграничават възможно най-много от

подписите на останалите потребители.

1.1.4 Он-лайн и оф-лайн СРП

В зависимост от начина на събиране на подписи, системите за разпознаване по подпис са статични (оф-лайн) и динамични (он-лайн). При първите подписът се привежда в цифров вид чрез сканиране или заснемане с фотокамера след като вече е бил положен върху хартия и по-нататък се работи с неговото полутоново статично изображение, докато при вторите подписите се полагат директно чрез дигитална писалка върху устройства като графичен таблет или сензорен екран (Tablet PC, умни телефони и т.н.), като по този начин автоматично се получава едновременно и статична, и динамична информация за подписа.

Производителите на такъв вид устройства се стремят да улеснят потребителите в най-голяма степен и да им предоставят възможност за писане по естествен за тях начин и с възможност за визуален контрол на написаното. Така например, някои устройства като графични таблети използват дигитална писалка с мастило (*inking pen*), с която се пише върху хартия, поставена върху повърхността на таблета. Тогава писалката пише по стандартния начин с мастило върху листа, но в същото време се създава и точно дигитално копие на подписа. При устройството Tablet PC пък се пише директно на екрана.

И така, би могло да се обобщи, че он-лайн СРП най-общо представят начина, по който се изписва подписа, докато оф-лайн СРП представят геометричната му форма.

Разбира се, всеки от методите има своите предимства и недостатъци. Поради по-голямата информация, която постъпва в динамичните системи в сравнение със статичните, те обикновено демонстрират по-високо качество [1]. Като недостатък би могло да се изтъкне, че за ефективната работа на он-лайн СРП обикновено е необходимо подписите да се полагат върху устройства от един и същ тип, въпреки че в последните години се обръща голямо внимание и на въпроса за т.нар. взаимозаменяемост на устройствата (*sensor interoperability*) [25]. Като основен недостатък на оф-лайн СРП се явяват по-дългото време и по-сложните методи, необходими за предварителна обработка на подписите (като премахване на шум от устройството и др.). Също така предизвикателство пред изследователите се явява и дебелината на следата на писалката. Предимство на динамичните СРП е наличието на динамична и времева информация, както и натиск, които се отчитат в момента на полагане на подписа от съответното устройство. Тъкмо тази информация за подписа е

изключително трудна за фалшифициране или с други думи, он-лайн подписите са значително по-трудни за подправяне от оф-лайн подписите. Затова и най-съществения недостатък на оф-лайн СРП е тъкмо лесната фалшификация на статичните подписи, тъй като тя се извършва само по формата им. В допълнение към недостатъците на оф-лайн СРП ще посочим и необходимото повече изчислително време за предварителната обработка на подписите.

И така, като обобщение на казаното дотук, он-лайн СРП се явяват по-надежден метод за разпознаване на потребител, произтичащ от възможността да се следи постъпващата динамична информация при полагането на подписите.

1.1.5 Приложение на он-лайн СРП

Броят на възможните области на приложение на он-лайн СРП постоянно нараства заедно с разработката на разнообразни, удобни за потребителя устройства за работа с ръкописен текст. Он-лайн СРП намират приложение в здравеопазването (защита на записи за пациенти), в банковия сектор (банкови операции с чекове и кредитни карти), както и в следните задачи:

- Включване в система (локална или отдалечена), криптиране на документи;
- Наблюдение – Автоматичното сравнение на он-лайн подписи може да се използва за следене и откриване на хора (напр. от черни списъци);
- Биометрични системи за криптиране – при тях подписът се използва за генериране на криптографски ключове [53];
- Транзакции на кредитни карти;
- Офиси без хартия, т.нар. зелени офиси: документите се подписват електронно, без да се отпечатват и по този начин се осигурява повсеместен достъп до тях и възможност за автоматична верификация. Това води до автоматизация на процесите и работното натоварване;
- Паспорти – при влизането в чужда държава, би могло да се изиска от пътника да се подпише върху графичен таблет, след което подписа му да се сравни с този, записан на магнитна лента в паспорта.

1.1.6 Видове подправени подписи в он-лайн СРП

За разлика от някои биометрични модалности, подписът може да бъде подправен с относителна лекота. Разграничават се следните типове подправени подписи: неумели (*random*), умели (*skilled*) и прости (*simple*). При първите фалшификаторът използва

собственият си подпис вместо този, който ще подправи, докато при вторите фалшификаторът е запознат с собствения подпис и динамиката на изписването му. При простите фалшификации пък, имитаторът изхождайки от името на потребителя, чийто подпис ще подправи, създава подправения подпис. На Фигура 1.1 са представени собственият и трите типа подправени подписи за потребителя "Десислава Николова".

Фигура 1.1 Собствен и три вида подправени подписа (неумел, умел и прост) за потребител

Степента на сходство на подправените подписи със собствените зависи от протокола за събиране на подписи, от мотивацията на фалшификаторите, от техните умения да подправят подписи, от това доколко са запознати с подписите и по какъв начин се упражняват, както и от времето, с което разполагат за трениране. Събирането на подправени подписи от хора, които нямат опит във фалшифицирането на подписи, както се случва на практика, е причина за отсъствието на качествени фалшификации. Фалшифицирането на динамиката при подписване е значително по-трудно отколкото имитирането единствено на формата на подписа, тъй като на практика фалшификаторът рядко има достъп до динамичните характеристики на подписа, както и до упражняването на различен натиск в различните елементи на подписа.

1.1.7 Етапи в създаването на СРП

Създаването на СРП (он-лайн или оф-лайн) преминава през следните етапи: събиране на подписи, предварителна обработка, извличане на признаци и избор на най-информативните сред тях, разработване на класифициращи правила и оценяване на вероятността за грешка при класификацията [1].

1) *Събиране на подписи* – Събирането на он-лайн подписи се осъществява чрез устройства, например графични таблети, PDA, Pocket PC, Tablet PC, умни телефони и др. Те се отличават с различни размери на активната площ, разделителна способност, скорост на предаване на данните към компютъра и чувствителност на натиск. При по-скъпите модели се отчита и наклона на

специалната писалка спрямо графичния таблет. Така, в зависимост от вида на използвания хардуер се измерват различни характеристики. Най-често това са: а) x, y координатите на върха на писалката; б) приложеният от писалката натиск; в) времеви отпечатък; г) ъгъл на азимута на писалката (ъгъла между проекцията на писалката върху плоскостта на таблета и оста x със стойности между 0° и 359°); д) ъгъл на наклон на писалката по отношение на повърхността на графичния таблет (със стойности между 0° и 90°).

2) *Предварителна обработка* – това е обработката, която се извършва, за да се подготвят подписите за следващия етап. Тя обикновено включва филтриране, за премахване на точките, непринадлежащи на подписа, транслация на подписа в дадена точка, ротация, целяща подравняването на подписа хоризонтално, а понякога и нормиране, за да се стандартизира подписа по отношение на размера.
3) *Извличане на признаци* – Те се пресмятат като функции от извлечените характеристики. Признаците би трябвало да позволят разграничаването на собствените от подправените подписи и да са подходящи за автоматично сравнение. Намирането на добро дискриминационно множество от признаци предопределя до голяма степен успеха на всяка система за разпознаване.
4) *Класификация* - Основната задача в областта на разпознаването на образи е построяването но класификатор. При нея се извършва причисляване на входните образци към един от множество от *N* класа. При он-лайн системите обикновено се решава задачата за верификация. Тогава даден участник заявява самоличност с полагане на подписа си и от системата се очаква да даде отговор дали подписът е собствен (заявената самоличност се приема) или не е (заявената самоличност се отхвърля). За целта избраният класификатор оценява сходството между вектора от признаци на подписа, който ще се разпознава и векторите от признаци на подписите на други лица. Различните техники за разпознаване по подпис са разгледани в раздел 1.2.
5) *Оценка на вероятността за грешка* - Качеството на биометричните системи се оценява в термините на оценки на грешката при вземане на решение (*decision error rate*). В качеството на оценки се използват FAR (*false accept rate*, дял на лъжливо положителните образци) и FRR (*false reject rate*, дял на лъжливо отрицателните образци). Първата от двете оценки отразява случая, когато биометричната система предоставя достъп на неоторизирано лице и се нарича

грешка от *II тип*. Втората оценка се отнася случая, при който биометричната система отказва достъп на оторизиран потребител до нея. Тя често се нарича и грешка от *I тип*. Разбира се, идеалният случай е когато оценките на FRR и FAR са близки до нула. Следва да се отбележи, че стойностите на FRR и на FAR са корелирани и всеки опит да се намали едната оценка неизменно води до увеличаването на другата. Отношението между двете грешки зависи от значението на ресурса, до който се иска достъп. В повечето случаи е за предпочитане грешката от втори тип да бъде намалена за сметка на грешката от първи тип.

1.1.8 Извличане на признаци и подходи за разпознаване по подписи

Признаците на подписи представляват функции от извлечените характеристики, а изборът на оптимално подмножество от признаци е важна стъпка в създаването на СРП. От първостепенно значение е извлечените признаци да имат невисоки изисквания към изчислителни ресурси и да водят до добро разпознаване с ниска стойност на FRR и FAR. За надеждно разпознаване по подпис признаците би трябвало да позволят подписите от различните потребители да се разграничават максимално, като същевременно да допускат известна вариация между подписите от един и същи потребител. Различните СРП се различават по използваните признаци и метода на разпознаване. Повече от 100 признаци са използвани в разработките в областта [3]. В зависимост от критерия, по който се групират, признаците за подписи се делят на [11]:

А) Според начина на събиране на подписите: статични и динамични

Б) Според обхвата: глобални и локални

В) Според начина на представяне: функционални и параметрични (скаларни)

Глобалните признаци са тези, които се отнасят до подписа като цяло, например дължина и височина, брой точки, наклон на средната линия на подписа спрямо оста x, средна скорост на подписване, време за полагане на подписа, дескриптори на Фурие за траекторията на подписа и др. Локалните признаци се пресмятат за всяка точка от траекторията на подписа. Такива са например координатите, скорост, натиск, наклон на писалката и др. във всяка точка от подписа. Подписите на даден потребител се отличават един от друг по дължина и затова при разпознаване, базирано на локални признаци е необходимо да бъдат сравнявани вектори с различна дължина (това най-често се извършва чрез метода на скрити Марковски модели или метода на динамично времево изкривяване [11, 17, 18]). В този аспект предимството на глобалните признаци

е, че при тях независимо от дължината на подписа се пресмятат фиксиран брой признаци. Някои глобални признаци могат да се използват като локални и обратно [11]. Сегментните признаци се пресмятат за всеки един от сегментите на подписа. Различни критерии за сегментация са посочени в [4], така например краят на сегмент и началото на нов сегмент биха могли да се определят от точките, в които скоростта по y стане равна на нула или от екстремните точки на самия подпис.

Според вида на използваните признаци, има два подхода за разпознаване на подпис, а именно функционален подход (базиран на признаци-функции) и параметричен подход (базиран на признаци-параметри) [1, 11]. Авторите от [1] заключават, че функционалният подход е по-скъп на изчислителни ресурси и следователно по-бавен, но пък значително по-точен. Когато се прилага функционалния подход, подписът се представя като времева функция, чиито стойности са множеството от признаци. Пример за признаци-функции са координатите (получават се директно от устройството), скорост и ускорение (може да се получат от устройството, а може и да се изчислят директно от координатите). При параметричния подход пък подписът се разглежда като многомерен вектор с компоненти глобални или локални признаци [7,8,9]. Някои от глобалните признаци по същество представляват функции от типа средна стойност, минимум, максимум на координатите, отместванията, скоростта и ускорението и др., други се определят като коефициенти от различни трансформации – Фурие, Уейвлет, Адамар, а трети могат да се извлекат от геометричния анализ на подписа (напр. височина и ширина на подписа, отношението между дължината и ширината, броя на характеристичните точки и т.н.). Възможен е и хибриден подход – комбинация от прилагането на тези два. Такъв комбиниран подход е представен в [3]. Проведеният експеримент показва, че качеството на този комбиниран метод е по-високо, отколкото ако се приложи всеки от двата подхода поотделно. Каквото и множество от признаци да е избрано, в някои работи са предложени методи за избор на подмножество от признаци за всеки участник, обосновавайки се на факта, че подписите на хората са уникални сами по себе си [11].

1.1.9 Трудности при създаването на СРП

Разпознаването по подпис е възприет метод за установяване на самоличност в множество приложения. Някои от тези приложения изискват високо ниво на сигурност и поради това е необходимо качеството и надеждността на дадена СРП да бъдат максимално подобрени преди тя да се пусне в експлоатация. Тъй като подписите

представляват изключително чувствителни данни, от гледна точка на тяхното съхранение в бази от данни би следвало да се осигури защитена среда, която да ги предпази от неправомерен достъп. В литературата са представени няколко начина за защитено съхранение на подписи [53]. Подписът има следните характерни особености, които също създават известни затруднения при създаването на надеждна СРП. На първо място следва да се отбележи, че представителите на подписите от един и същи потребител са подобни, почти еднакви, но не и идентични (съществува голяма вътре класова вариация , Фигура 1.2), както и че последователно положени подписи от един и същ потребител биха се различавали по мащаб и по ориентация. Върху качеството на подписите оказват влияние и различните условия, при които те се полагат. Например набързо положения подпис в изправена поза би се различавал от подписите, положени докато участникът седи. Значение би оказала и употребата на необичайна писалка, с която потребителят не е свикнал, както и моментното психологическо състояние на участника. Трудностите при разпознаването произтичат от късата дължина на някои подписи, която би могла да разкрие малка информация, а оттам да доведе до неточното им разпознаване. Също така, потребители с подобни или еднакви имена би могло да имат сходни подписи по геометрична форма [11].

Трудности произтичат и от процеса по набавянето на собствени и подправени подписи. Обикновено СРП разполага с ограничен брой подписи. Хората се отегчават, ако от тях се поиска да положат повече от пет броя подписа. При прилагането на някои методи за разпознаване (напр. скрити Марковски вериги) малкият брой примери за обучение не дава добри резултати. Основните техники в областта на разпознаването на образи могат да се приложат, ако са налични достатъчен брой примери от всички класове (в случая на верификация са два класа). На практика, обаче, само няколко собствени подписа са налични. Труден е процесът по събиране на подправени подписи, необходими на етапа на тестване на точността на СРП. Създаването на тестова база от данни от подписи, която да е представителна за реални приложения е тежка задача, поради трудностите, които обичайно се срещат при набирането на доброволци, които да се съгласят да представят известен брой подписи и то в различни сесии. Обикновено две са достатъчни. Необходимостта от набиране на подписи в различни сесии се обуславя от допускането, че по този начин биха се събрали по-достоверни представители на подписите, а не само такива от една сесия, за които се предполага, че ще са изключително близки, без много вариации. Повечето от хората не биха се съгласили подписите им да се съхраняват в бази от данни и да се предоставят за

фалшифициране.

Фигура 1.2 Вътрекласова вариация между подписите на потребител

Като обобщение на изложеното дотук, може да се заключи, че задачата за разпознаване по подписи е достатъчно актуална и значима, което предопределя и значителния интерес от страна на научната общност и частни компании към разработването и прилагането на ефективни методи за избор на признаци и техники за разпознаване за създаването на точни и бързо действащи СРП.

1.2 Критичен обзор

В тази част е извършен преглед на разработките в областта на разпознаването на подписи от последните години. Обърнато е внимание на обзорните статии по темата и са разгледани различни подходи за решаване на задачата, свързани с настоящата дисертация. Акцентирано е върху разработките в областта на разпознаването на он-лайн подписи, тъй като тъкмо те ще бъдат обект на изследване в работата.

Големият обем работа в областта е отразен в няколко фундаментални и значими обзорни статии. През 1989 г. е издадена първата от тях с автори Plamondon и Lorette [1]. В следващият й, актуализиран вариант [15] от 1994 г. е направен преглед на он-лайн и оф-лайн системите за разпознаване на подпис, като вече са включени и методи за разпознаване, базирани на невронни мрежи. През 1997 Nalwa [17] прави проучване върху разработките до момента в областта на он-лайн подписите, а през 2000 г. излиза обширната статия [2], в която изчерпателно са разгледани съвременните разработки по темата. Следва труда на Impedovo [11], включващ повече от 300 заглавия в библиографската справка. През 2006 Gupta извършва критичен обзор по темата он-лайн подписи [4]. Последната до момента обзорна статия в областта на он-лайн подписите от 2008 г. е дело на колектив, воден от Fierrez-Aguilar [16].

През последните 30 години са разработени голям брой методи и модели за верификация на он-лайн подписи, които се основават най-общо на следните методи: статистически подходи, динамично времево изкривяване (*Dynamic Time Warping*), скрит Марковски модел (*Hidden Markov Model*), невронни мрежи и комбинирани подходи [24].

В настоящата част изброените методи са дискутирани и са представени конкретни системи за разпознаване по подпис, проектирани върху тях. За всяка от системите са

посочени използваните видове признаци и метода за разпознаване, вида на използваните фалшифицирани подписи, базата от данни, върху която са проведени експериментите и получената оценка на системата.

Традиционните **статистически методи** за класификация като напр. дискриминантния анализ са построени на базата на Бейсовата теория за вземане на решения [13]. От съществено значение за тези методи е, че трябва да има направено допускане за съответния вероятностен модел, за да може да се изчислят апостериорните вероятности, по които ще се извършва класификацията. Недостатък на тези модели е, че те работят добре единствено при условие, че направените допускания са изпълнени. Тяхната ефективност зависи до голяма степен от различните допускания и условия, върху които са построени моделите. Изследователите би трябвало добре да познават както данните, така и възможностите на модела, за да могат да го приложат успешно. Прилагането на статистически методи [30] води до бързи изчисления, но и до по-неточни резултати в разпознаването. За сравнение, невронните мрежи са по-точни за сметка на по-дългото време, което е необходимо за обучение.

Тъй като обикновено подписите от един и същ потребител са с различна дължина, то не е възможно да се приложи линеен метод като метода на Евклидовото разстояние за сравнение и се налага прилагането на нелинейни подходи. Такива са например методите на динамично времево изкривяване и скрит Марковски модел [5], базирани на локални признаци, които преди да намерят приложение като подход за верификация на он-лайн подписи, се прилагат успешно за разпознаване на говор. Обикновено резултатите, получени от прилагането им надвишават тези, получени от методи с използване на глобални признаци [11]. Проектирането на СРП върху локални и глобални признаци води до повишаване на точността и увеличаване на скоростта на разпознаване.

Методът на **динамичното времево изкривяване** е скъп на изчислителни ресурси, но въпреки това намира широко приложение в разпознаването на говор поради голямата си точност, а в последните години – и в разпознаването по подпис. При него се извършва сравняване на подписи (последователности от точки) с различна дължина. Търси се подравняване между точките в двете последователности, така че сумата от разликите между всяка двойка подравнени точки е минимална. За целта е необходимо да се разгледат всички възможни подравнявания. Техниката на динамичното времево изкривяване се базира на динамичното програмиране. Чрез прилагането й, постъпилият подпис за тестване се подравнява, и повторно се дискретизира нелинейно по отношение

на запазения в базата данни образец на подписа за съответния потребител.

В [18] е разработен метод за разпознаване по подписи, положени върху графичен таблет. Извличаните локални признаци са пространствени и времеви. За изчислението на пространствените признаци за всяка точка от подписа се пресмятат следните локални признаци: *x* и *y* координатни разлики между текущата и следващата точка от подписа, полутонови стойности в околност 9x9 на точката, стойност на у координатата след извършване на повторно дискретизиране (*resampling*), синус и косинус на ъгъла, който правата през текущата и следващата точка сключва с оста *x*. За изчислението на времевите признаци се пресмятат локалните признаци абсолютна и относителна скорост във всяка точка от подписа. Единственият използван глобален признак е брой щрихи в подписа. Прилаганите методи за предварителна обработка са повторно дискретизиране, изглаждане и мащабиране по отношение на размера. Предложеният метод е тестван върху две бази от данни за подписи. Първата се състои от 520 собствени подписа и 60 умели фалшификации, събрани от 52 участника, а втората от 1232 собствени и 60 умели фалшификации, събрани от 102 участника. Проведени са експерименти с прилагане на общ праг и на специфичен за всеки участник праг. Най-добри резултати (2.8% FAR и 1.6% FRR) са получени с прилагане на специфичен за участниците праг. Прилагането на общ праг води до стойности 3.3% FRR и 2.7% FAR.

Yanikoglu и Kholmatov [6] представят система за верификация на подпис, с която печелят през 2004г. състезанието за проектиране на СРП SVC2004 (*Signature Verification Competition*) [5]. При извършването на верификацията на даден подпис, последователно се изчисляват разстоянията между него и всеки един от подписите на съответния потребител. За целта е необходимо предварителното подравняване, предискретизиране на подписите чрез метода на динамично времево изкривяване. Разстоянията между подписа за тестване и (1) най-близкия, (2) най-далечния и (3) еталона на подписа на съответния потребител се нормират и заедно формират вектор с дължина, равна на три. Така полученият вектор се използва за класификация. Авторите извършват експерименти с три различни класификатора – линеен, метод на опорните вектори и Бейсов класификатор. Базата от данни се състои от 94 участника и 1134 подписа (собствени подписи и умели фалшификации). За събиране на умелите фалшификации авторите са разработили модул, в който полагането на собствен подпис се заснема и фалшификаторът разполага с време да се упражни в имитирането на подписите, които първо е изгледал във видеото. Най- добрите резултати са получени за линейния класификатор: 1.64% FRR и 1.28% FAR.

Скритият Марковски модел е моделът, който обикновено проявява най-добро качество в областта на разпознаването по подпис [45]. Предимството му се състои в това, че позволява известни вариации в подписването като същевременно се обръща внимание и на индивидуалните характеристики на подписите. По същество, скритият Марковски модел моделира двойно стохастичен процес, представляващ Марковска верига с краен брой състояния и множество от случайни функции за преход, всяка от които е свързана с изхода на едно от състоянията. В дискретен момент от време t_i, процесът се намира в едно от състоянията и според случайната функция, съответстваща на текущото състояние, попада в следващото състояние.

Ortega-Garcia и колектив [39, 45] използват скрит Марковски модел с 4 състояния и 8 комбинации за състояние, за да моделират локалните признаци. При полагане на подписите се извличат стойностите на *x*, *y* координати, натиск, азимут и ъгъл на наклон на писалката. С тяхна помощ се изчисляват три локални признаци, първата и втората им производна и така общият брой признаци е 24. Така, ако даден подпис съдържа 1000 точки, ще се генерират 24000 стойности. Тези стойности се нормират така, че да имат средна стойност, равна на нула и стандартно отклонение, равно на единица. Базата данни е MCYT-50, съдържаща подписите на 50 участника с 15 собствени и 15 подправени подписи от всеки. Прилагането на общ праг за всички участници води до 4.83% EER, докато прилагането на специфичен за всеки участник – до 0.98% EER. В [26] същите автори подобряват модела и добавят още един локален признак. Създаденият модел е със 2 състояния, 32 комбинации за състояние с резултат 0.74% EER за умели фалшификации и 0.05% EER за неумели фалшификации върху част от базата данни за SVC2004 [5], съдържаща 3625 собствени, 3625 умели фалшификации и 41760 неумели фалшификации от 145 участника. Представената СРП е класирана на второ място в състезанието SVC 2004 [5] след тази на Yanikoglu и Kholmatov [6], която се базира на метода на динамичното изкривяване. В [36] авторите от колектива на Ortega-Garcia демонстрират, че при наличието на достатъчен брой подписи за обучение, резултатите, получени от прилагането на скрития Марковски модел са по-добри от тези, получени от прилагането на метода на динамичното изкривяване на Yanikoglu и Kholmatov [6].

Невронните мрежи са популярни класификатори в областта на разпознаването на образи поради възможността им за самообучение от данните. През последните години намират широко приложение в разпознаването на подписи. Причината за това на първо

място е тяхната гъвкавост при промяна в данните (в случая вариацията сред подписите на всеки един потребител). Други причини са възможностите, които предлагат (моделиране на сложни функции) и лесното им прилагане (тъй като се обучават чрез примери, за изследователя остава единствено да набави достатъчно представителни данни и да моделира невронната мрежа). Най-популярните невронни мрежи са многослойните перцептрони, обучавани по алгоритъма за обратно разпространяване на грешките. Този вид мрежи са известни с високото си качество при класификация в системите за разпознаване по подпис [20,35]. Те се представят добре и когато броят на примерите за обучение е малък [34].

Най-общо съществуват два вида класификация: дихотомия (с два класа, верификация) и мултикласова (с повече от два класа, идентификация). Те определят и два подхода при прилагането на невронни мрежи за подписи. При *първия* подход за всеки участник се използва отделна невронна мрежа с един неврон в изходния слой, приемащ стойности между 0 (фалшифициран) и 1 (собствен) подпис. Тъй като размерът на невронните мрежи в този случай е малък, то използването на отделна невронна мрежа за всеки отделен потребител води до бързо обучение. При въвеждането на нови негови подписи или при промяна на подписа и въвеждането им отново се провежда бързо ново обучение. Недостатък се явяват по-големите изисквания към ресурси и данните за обучение (задължително е наличието на фалшифицирани подписи). При *вторият* подход се използва една обща невронна мрежа за всички участници. Неудобството е, че тя трябва да се обучи наново при добавянето на нов участник и тъй като обучението е свързано с размера на невронната мрежа, следователно колкото повече са потребителите, толкова повече са необходимите възли в мрежата и толкова по-дълго е времето за обучение. Предимство е, че не са необходими фалшифицирани подписи. Обучението и при прилагането на двата подхода се провежда в оф-лайн режим.

Авторите Baltzakis и Papamarkos [20] представят нов метод за идентификация на оф-лайн подписи, базиран на класификация с малки и лесно обучаващи се класификатори на две нива. Признаците се разделят на три групи: глобални (16 броя), мрежови (96 броя) и текстурни (48 броя). В първото ниво се използват три многослойни перцептрона - по един за всяка група от признаци. Всяка от трите невронни мрежи съдържа по един неврон в изходния слой, който приема стойност между 0 и 1, измерваща мярката на сходство между разглеждания подпис и множеството от подписи за обучение за съответния потребител. Обучението се извършва чрез алгоритъма

ALOPEX, който е вариант на алгоритъма за обратно разпространяване на грешките и за разлика от него не позволява на невронната мрежа да попадне в локален минимум на оценъчната функция. Във второто ниво се използва невронна мрежа с радиални базисни функции, която комбинира резултатите получени на първото ниво и взима финално решение дали разглежданият подпис е собствен или не. Предложената СРП е мащабируема и скалируема, тъй като включването на нови подписи не изисква обучението на цялата система, а само на малка част от нея. Друго нейно предимство е, че при проектирането на системата не е необходимо предварително познаване априори на броя на потребителите в системата.

В [40] се извършва обучение на две невронни мрежи. Едната невронна мрежа се обучава за формата на подписите, а другата за разпределението на натиска във всички точки от подписа. Предварителната обработка на подписите включва ротация и мащабиране, така че подписите на даден потребител да се подравнят с първия му подпис. Чрез прилагането на метода на динамичното времево изкривяване се постига броят на точките във всеки от четирите подписа, събирани от потребители да е равен. Невронните мрежи са многослойни перцептрони с един скрит слой, обучени с алгоритъма за обратно разпространяване на грешката по метода на спускане по градиента с адаптиращи параметри инерционен момент и степен на обучение. Входът на първата невронна мрежа са x координатите, а изходът – съответните y координати. По време на верификацията, се изчислява Евклидовото разстояние между y координатите, получени като резултат от невронната мрежа и средните стойности на y координатите за съответния потребител. Ако разликата е по-малка от 15 %, се продължава с верификацията с втората невронна мрежа. Предложената система е тествана с 30 подписа от 3 потребителя и неумели фалшификации. Получените резултати са 17% FRR и 0% FAR. По-късно същите автори подобряват предложената от тях система [31]: 1.33% FRR и 0% FAR при тестване с 150 подписа от 5 потребители с неумели фалшификации. Добрите резултати се дължат най-вече на неизползването нито на умели фалшификации, нито на публични бази от данни. Първата невронна мрежа приема за вход x и y координатите, а в изходния слой са стойностите на разпределението на натиска, а втората приема за вход x и y координатите, а изходния слой е профила на скоростта при изписването на подписа. Обучението е чрез алгоритъма на Levenberg-Marquardt .

В [52] е предложен подход за преодоляване на практическите трудности, които обичайно се срещат при създаването на СРП – вътрекласовата вариация сред подписите

на даден потребител и набирането на умели фалшификации. Отделя се специално внимание на т.нар. стабилни части от подписа. Авторите дефинират два типа стабилни части: големи стабилни компоненти и локални компоненти. Невронната мрежа се обучава с изкуствено получени фалшификации на подписите чрез премахването на големи стабилни части от тях. По този начин класификаторът се обучава да открива тези стабилни части в подписа. Чрез присвояването на по-високи тегла на възлите, съответни на локалните стабилни части, невронната мрежа се научава да обръща по-голямо внимание на тях. Проведен е експеримент с подписи от 9 потребителя. За всеки един от тях се генерират две извадки от подписи. Първата съдържа собствените подписи и неумели фалшификации, а втората съдържа собствените и фалшификации, получени чрез премахването на големи стабилни части от тях. Тестовата извадка съдържа собствени подписи, не участващи в обучаващите извадки, умели и неумели фалшификации. Общият брой на подписите е следният: 350 собствени, 158 умели и 230 неумели фалшификации. Резултатите показват, че така генерираните фалшификации биха могли да се използват при липса на умели фалшификации.

В [51] са тествани различни архитектури на невронни мрежи за верификация на подпис. Най-добрият получен експериментален резултат е 2% FAR и 1.3% FRR за многослоен перцептрон с един скрит слой. При създаването на многослоен перцептрон с два скрити слоя се повишава както времето за обучение, така и грешката 2.5% FAR и 1.3% FRR. Във всичките експерименти са използвани по 5 подписа от всеки потребител за обучение и 41 глобални признаци на входа.

Добри резултати са получени с използването на Фурие дескриптори в ролята на глобални признаци за он-лайн подписи [27]. Получава се компактно представяне на подписите с различна дължина чрез фиксиран брой глобални признаци. Авторите са използвали нормираните коефициенти от трансформацията на Фурие, приложена върху *y* профила на подписите. X профилът на подписа не се разглежда за простота, тъй като той не съдържа особено разграничаваща информация. Резултатите, получени за умели фалшификации за базата от подписи SUsig Visual са EER 6.2%, а за MCYT-100 – 12.1%. Авторите прилагат и сливане (fusion) на метода с този за разпознаване на подписи чрез DTW [6] и получават намаляване на EER с 25%.

В статията [37] е предложена система за верификация на он-лайн подписи за Tablet PC и личен дигитален помощник (PDA) на две нива - Гаусов смесен модел за глобални признаци и дискретен скрит Марковски модел за локални признаци. Даден подпис се счита за собствен, ако премине успешно през двете нива на верификация. Необходими

са само пет собствени подписа за всеки потребител. Извличаните глобални признаци са два типа: пространствени (ширина, височина, обща дължина, брой на щрихите, брой на сегментите и динамични (средна скорост, максимална скорост, среден натиск и максимална стойност на изменението на натиска за две последователни точки от подписа, времетраене и отношение на времето, в което писалката допира графичния таблет към общото време). С това се постига бързина и ефикасност, тъй като в процеса на обучение не се използват подправени подписи. Затова и цитираният подход за верификация на подпис може реално да се приложи на практика, тъй като в реалния случай СРП не разполагат с подправени подписи. Допълнително предимство е, че необходимите подписи от всеки участник са само пет, а и в действителност потребителите не са склонни да се подписват голям брой пъти. За всеки потребител се изчисляват независимо един от друг и двата модела. Чрез Гаусовият смесен модел се оценява разпределението на глобалните признаци (времетраене, средна скорост) и чрез него бързо се отхвърлят фалшификациите на базата на пространствените признаци. Точността на разпознаване на системата е 93.3%.

Комбинираните подходи за класификация са област на активни разработки през последните години. Ансамбъл от класификатори може да се образува от няколко различни класификатори, от няколко различни невронни мрежи, от един и същ модел невронна мрежа, обучена с различни алгоритми или различни стойности на началните тегла. Начин за създаване на комбиниран класификатор е чрез създаването на мултиекспертна система, състояща се от няколко експерта (класификатори), всеки специализиран в решаването на конкретен проблем (напр. обучени с признаци от различни групи). Ефикасността от прилагането на тази стратегия е отразена в [21,22]. Комбинирането на класификатори може да се извърши паралелно или последователно (многослойни класификатори).

Целта на комбинирането на няколко класификатора е подобряването на качеството пред това на отделните класификатори, а също и избягването на евентуалната грешка от работата на единствен класификатор. Доказано е теоретично, че качеството на ансамбъл от класификатори надвишава това на всеки един от класификаторите, разглеждан отделно, при положение, че изходите от класификация от всичките класификатори са некорелирани [3, 46]. Основата, върху която се обуславя мултиекспертния подход идва от допускането, че чрез прилагането на комбинация от резултатите от няколко експерта е възможно да се компенсира евентуалната слабост на някой/ някои от експертите [46]. Използването на допълващи се експерти води до по-добра комбинирана система.

Съществуват разработки в областта на разпознаването по подпис, в които се прилага хибриден подход, при който част от класификаторите оперират с глобални признаци, а другата част – с локални признаци [23]. Тези мултиекспертни системи биха могли да комбинират нови методи с вече доказали се методи като динамично времево изкривяване и скрити Марковски вериги, както е направено например в [6]. В [23] е представена мултиекспертна система за верификация в три последователни нива. Първият експерт използва само един общ признак – контурите на подписа. Неговата задача е да идентифицира неумелите фалшификации. На следващото ниво, вторият експерт получава само тези подписи, които не са отхвърлени на първото ниво (това са собствени подписи и умели фалшификации) и има за задача да открие умелите фалшификации, на базата на признака – региони от подписа с голям натиск. Класификаторите и на двата етапа са многослойни перцептрони, обучени с алгоритъма за обратно разпространение на грешките. Същевременно и на двата етапа се изчисляват критерии за оценка на надеждността на класификацията от съответния експерт и в случай на несигурност, разглежданият подпис се препраща за класификация на последния експерт, който взима финалното решение, взимайки предвид оценките за надеждността на класификацията от предишните етапи. Системата е тествана върху база от данни от 1960 подписи от 49 участника. Експериментите показват по-добри резултати от тези, получени с използването само на един етап и по-голямо множество от признаци.

В [41] са представени три различни групи от глобални признаци за класификация на оф-лайн подписи. Те са (1) хоризонтална и вертикална проекция (които дават статистическа мярка за разпределението на точките от подписа) и разпределение на крайните точки по горната (2) и долната (3) част на подписа, съответно горен и долен профил. Използвани са три класификатора с прави невронни мрежи (НМ с праволинейно разпространение на сигнала) по един за всяка група признаци. Обучени са чрез стандартния алгоритъм за обратно разпространение на грешката. Тъй като времето за обучение е дълго, обучението се провежда в оф-лайн режим. Мрежите съдържат два скрити слоя, първият съдържа толкова възли колкото са тези на входа, а вторият съдържа 2/3 от този брой. Резултатите се комбинират в конекционисткия модел ADALINE, който има толкова неврони в изходния слой колкото е и броя на потребителите, а именно 10. Всеки един от участниците се подписва 15 пъти. Неумелите фалшификации са общо 100 на брой. Статистическите резултати показват, че комбинирането на резултатите води до намаляване на броя на грешните

класификации.

Системата за разпознаване на подписи [9] е базирана на сливането на два машинни експерта като единият е базиран на глобални признаци с класификатор за измерване на разстоянието, докато вторият е базиран на локални признаци и подписите се моделират със скрит Марковски модел. Експериментите се провеждат върху базата от данни MCYT за умели и неумели фалшификации. Показано е, че експерта, базиран на локални признаци има по-добро представяне от другия във всички тестови сценарии. Също така е демонстрирано, че предложените две системи дават допълваща се информация и чрез комбинирането на изходите им чрез правилото за сумиране се получават по-добри резултати.

Две системи за разпознаване на он-лайн подписи въз основата на локални и на глобални признаци са представени в [3]. Взима се решение на базата на сливане на решенията от двете системи. Глобалните признаци са 100 на брой, представени параметрично и разпознаването се извършва чрез класификатор с Парзенови прозорци. Локалната информация се извлича като 14 дискретни времеви функции на различни динамични характеристики и се разпознава чрез скрити Марковски модели. Базата данни, върху която се тества предложения метод, е публично достъпната MCYT [330 участника и общо 16500 подписа, сред които и неумели и умели фалшифицирани подписи]. Проведени са експерименти за избор на признаци (заради големия им брой - 100), чрез ранжиране според вътре-вътрекласовата им дисперсия. Избрани са първите 40 признака. Показано е експериментално, че системата, базирана на локални признаци е с по-добро качество от тази с глобални признаци при наличие на достатъчно на брой примери за обучение. Обратно, показано е, че в случай на малък брой примери за обучение системата с глобални признаци се представя по-добре. Двете представени системи дават допълваща се информация при разпознаването, което спомага за сливане на решенията им чрез функциите максимум и сумиране. Експерименталните резултати са обещаващи и показват за умелите фалшификации EER със стойност между 5 и 7%, получени при множество за обучение от 5 подписа и EER със стойност между 1 и 2% за умелите фалшификации, получени при множество за обучение от 20 подписа. Резултатите, получени за неумели фалшификации са EER със стойност между 1 и 1.5% при множество за обучение от 5 подписа и EER около 0.5% при множество за обучение от 20 подписа.

В литературата могат да се открият данни за приложение на метода на *k*-най-близки съседи в задачите за разпознаване по подпис, но главно за оф-лайн подписи. Така

например, в [106**]** авторите използват база данни за оф-лайн подписи, съдържаща 960 подписа от 40 потребителя. Подписите се нормират и мащабират до 102x64 пиксела, след което данните за тях се обработват чрез PCA и линеен дискриминантен анализ, като по този начин се формират признаци за класификатора по *k*-най-близки съседи. Чрез крос-валидация се намира стойност на оценката на точността 97,47%.

В статията [107] са представени резултатите от прилагането на класификатор SVM (метод на опорните вектори) и класификатора по *k*-най-близки съседи по глобални признаци от дискретна Радон трансформация. Базата данни за оф-лайн подписи се състои от 2250 подписа (собствени и умели фалшификати) от 75 потребителя. С прилагане на класификатор SVM се постига точност около 80%, а с kNN класификатор – 70%.

Авторите в [108] извличат глобални признаци от 24 собствени и 24 умели фалшификати на оф-лайн подписи за всеки потребител от базата данни. С прилагане на класификация по *k*-най-близки съседи получават точност 78% за верификация и 93% за идентификация.

Извод: Текущите изследвания и разработки в областта на разпознаването на подписи са насочени основно към създаването на комбинирани подходи и по-добър избор на признаци, които да разграничават надеждно собствените от фалшифицираните подписи.

1.3 Цели на дисертационната работа

От направения преглед на изследванията и разработките за верификация на потребители по он-лайн подписи и увеличаващата се потребност от такива разработки се определя основната цел на дисертацията:

Разработване на нов комбиниран метод за разпознаване на потребители по он-лайн подписи и реализирането му като софтуерна система.

За изпълнение на поставената цел е необходимо да се решат следните задачи:

- Организиране на база данни от подписи;
- Извличане на известни и нови признаци;
- Избор на оптимални подмножества от признаци;
- Изследване на различни модели на невронни мрежи за верификация като елементи на комбиниран класификатор;
- Провеждане на експерименти за оценка на точността на класификация и сравнение с известни класификатори с използване както на собствена база данни за подписи, така и с публично достъпна такава**;**

- Разработка на софтуерно приложение за разпознаване на подписи с основни функции: събиране на собствени и подправени подписи, визуализация на подписите, извличане на признаци, избор на признаци и верификация по тях.

Особеностите на задачата налагат определени ограничения, съобразяването с които прави някои от стандартните подходи не особено подходящи. Тези особености се отнасят преди всичко към количеството на експерименталните данни, използвани за обучение и тестване на класификатора и характера на използваните признаци. За разлика от други задачи тук не е възможно събирането на голям брой образци от подписи на евентуалните потребители. Изискването за многократно подписване е неприятно. Броят на желаещите намалява значително и ако се иска неколкократно повторение на сесиите. Добавяйки и нежеланието на повечето хора да предоставят подписите си за експерименти, страхувайки се от злоупотреба, броят на събраните подписи се свежда до няколко десетки. Сравнително по-големи бази от подписи могат да бъдат събрани в университетите. Още по-голямо затруднение представлява събирането на умели фалшификати, необходими за повишаване на точността на класификатора.

Изборът на признаци също не е тривиален по две причини. Първата е свързана с малкия брой опитни данни. От теорията на разпознаването на образи е известно, че при фиксиран брой опитни данни увеличаването на броя на признаците не само не води до повишаване на точността на разпознаването, но я и намалява. Това означава, че ако при малък брой данни имаме голям брой признаци, те трябва да се намалят, т.е. да се избере подмножество, което да бъде достатъчно информативно. Намаляването на броя на признаците е свързано и с намаляване на изчислителното време. Втората причина при избора им е необходимостта от разбираем физически смисъл. Това се налага при съдебни експертизи, където резултатът от машината не се приема от съда за доказателство, ако зад него не застане експерт. (Обяснението е, че от машината не може да бъде търсена отговорност).

Независимо от тези сериозни практически ограничения, към всеки класификатор се предявяват изисквания за висока точност на разпознаване, изразяваща се в малък процент на двата вида грешки: дял на лъжливо положителните образци (FAR) и дял на лъжливо отрицателните образци (FRR). Допълнителна особеност в това отношение е различната цена на двете грешки, която трябва да бъде отчетена от класификатора. Много по-сериозна е първата грешка, която може да доведе до тежки последици за

институцията, докато при втората ще бъде необходимо само полагането на допълнителен подпис.

Заедно с формулираните цел и задачи на дисертацията ще бъдат потърсени отговори и на следните въпроси.

1) Съществува ли конкретно, специфично подмножество от признаци за всеки потребител, отразяващо особеностите на неговия стил на подписване?

2) Има ли съществена разлика между резултатите, получени при верификацията с общо множество от признаци и индивидуално подмножество за всеки участник?

3) Ще се понижи ли точността на верификация с използването на умели фалшификати в качеството на отрицателни примери?

1.4 Резултати в Глава 1

1. Представена е биометричната модалност „подпис“, начините за получаване и възможните приложения.
2. Направена е характеристика на използваните признаци, етапите на изграждане на СРП и съпътстващите ги трудности.
3. Представени са основните методи за разпознаване: статистически, динамично времево изкривяване, скрити Марковски вериги, невронни мрежи, комбинирани подходи. Посочени са съответните литературни източници и данни за ефективността на методите.
4. Формулирана е целта на дисертацията и съпътстващите изпълнението ѝ задачи.

Глава 2. Избор на признаци за онлайн подписи

2.1 Предварителна обработка

Една от основните задачи в разпознаването на образи е изборът на признаци, по които да се извършва разпознаването. Те се избират от изследователя в зависимост от неговия опит, а също и от спецификата на разпознаваните обекти. Предварително не е известно кои са признаците, които биха довели до задоволително разпознаване и затова обикновено изследователят разглежда възможно най-голям брой признаци, стремейки се по този начин да не пропусне някой важен за разпознаването признак. Същевременно, ефективността на системата за разпознаване и изискванията към изчислителните ресурси зависят в голяма степен от броя на признаците [12]. Затова колкото е по-малък той, толкова е по-голяма скоростта на верификация [13]. В литературата е известен т.нар. *peaking phenomenon*, термин, с който се обозначава ситуацията, при която добавянето на нови признаци води до увеличаването на информацията, но също така се отразява неблагоприятно на точността на класификатора. Така например нека два признака водят до някаква точност, напр. 80%, добавянето на трети води до увеличаване на точността напр. 85%, но добавянето на четвърти я намалява до 83%. От друга страна, ако съществува подмножество от два признака, което води до точност от 90% , задачата е то да се открие.

Така възниква и задачата за избор на най-информативните признаци, решаването на която се състои в минимизиране на признаковото пространство, така че да се премахнат признаците, които са носители на излишна или повтаряща се информация.

Бихме могли да формулираме следните критерии към признаците, които да участват в множеството от признаци за он-лайн подписи:

1. Да бъдат извличани и пресмятани бързо и лесно;
2. Ако по някакъв начин станат достъпни от неоторизирани лица, да не позволяват лесното пресъздаване на подписа чрез тях;
3. Броят им да бъде достатъчно малък, за да не се забавят по-нататъшните изчисления и достатъчно голям, за да осигури надеждна верификация;
4. Желателно е да имат разбираем физически смисъл, което ще позволи установяването на фалшиви подписи от експерт в случай на съдебна експертиза.

Наред с избора на модел и вида на класификатора избраните признаци играят много важна роля за постигането на успех на всяка СРП.

2.1.1 Предварителна обработка на данните от подпис

Както е известно, според вида на използвания хардуер (графичен таблет, PDA и др.) за събиране на подписи, се измерват различни характеристики за всяка точка от траекторията, като най-често това са *x* и *y* координатите на върха на писалката, натиска, времеви отпечатък, азимут на писалката и ъгъл на наклон на писалката по отношение на повърхността на таблета. Данните за тези характеристики са в "суров" вид. С прилагането на някои операции към тях (т.нар. предварителна обработка), стойностите на данните се трансформират, за да може да се улесни извличането на признаците от тях. Нуждата от конкретни операции, извършвани в етапа на предварителната обработка, се обуславя основно от протокола за събиране на данни и от спецификата на извличаните признаци. Така например, в случай че интерфейса за събиране на подписи не налага ограничения за размера на полето, в което участниците се подписват, в някои задачи би било удачно да се приложи мащабиране, така че подписите от всичките участници да имат приблизително равна дължина (напр. ако се използват методи за сравняване на подписи от типа на динамично времево изкривяване и др.). Разбира се, това неизменно довежда и до загуба на ценна информация относно стила на подписване, съдържаща се в размера на подписа и времетраенето му. Това е и причината, поради която се въздържаме от прилагането на мащабиране в етапа на предварителната обработка. Също така не извършваме премахване на шум, понеже устройството, с което се събират подписите, а именно графичен таблет, не допуска наличието на шум (за разлика от сканирането чрез скенер например). Тъй като при събирането на данни от графичен таблет, се записва и информация за точки, получени дори и ако писалката е близо до повърхността на таблета, без да я докосва, се генерират точки от подписа, за които стойността на натиска е равна на нула. Тези области се наричат области с нулев натиск (*zero pressure regions*). Тъй като те не принадлежат на траекторията на подписа, е необходимо да се премахнат от по-нататъшно разглеждане. Възможно е да се генерират и щрихи, които не съдържат точки. Те също се изтриват. Високата скорост на предаване на данните (максимум 200 точки в секунда) е причината, поради която при по-бавна скорост на подписване се генерират множество точки с равни координати, но разбира се с различни стойности на времевия отпечатък, натиска и т.н. Всичките тези данни се запазват, тъй като изтриването на данните за точките с равни координати би довело до загуба на важна информацията, свързана със скоростта на писане.

Координатите *x* и *y* от координатната система на таблета (*ink coordinate system*) се

наричат *himetric units* [97] и приемат високи стойности в [0,7999]x[0,5999]. Това е така, тъй като е необходимо разрешаващата способност на графичния таблет да бъде висока, което води до високо качество на въведената от дигиталната писалка информация. Първата извършвана операция е *(1) трансформиране* на тези координати към x и y координатите от координатната система на приложението със стойности в [0,1279]x[0,799]. Това се извършва автоматично от метод, достъпен от пакета за разработка на приложения Microsoft Tablet PC.

При събирането на подписи с графичен таблет има възможност подписите от даден участник да не са подравнени по отношение на оста *x*, а да са завъртени на определен ъгъл. Това от своя страна налага нуждата от извършването на *(2) ротация*. Другата извършвана операция е *(3) транслация* на подписите от всички участници в избрана точка от координатната система на приложението. Тя се извършва, тъй като е необходимо при извличането на признаците, координатите на точките от подписа да имат положителни стойности, а при извършване на ротацията е възможно някои от тях да придобият отрицателни значения. По-долу в текста е направено описание на прилаганите две трансформации ротация и транслация.

Ротация на подпис на определен ъгъл

За извършването на ротацията на подпис се прилага подхода на собствените вектори v_1 и v_2. Първоначално се пресмятат съответните собствени стойности λ_1 и λ_2 на ковариационната матрица Σ_{xy} .Максималната собствена стойност $\lambda_{\max} = \max(\lambda_1, \lambda_2)$ показва оста на най-голямата промяна, а ъгъла θ, който подписа сключва с оста x може да се намери от компонентите на съответният й собствен вектор $v_{\max}$.

$$\theta = arctg(\frac{v_{\max 2}}{v_{\max 1}}) \qquad (2.1)$$

За даден подпис S с n на брой точки, представен във вида:

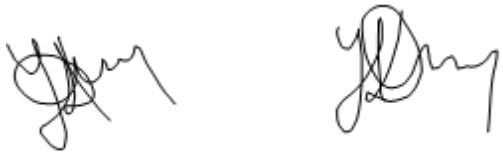
$$S = \{S_i(x_i, y_i), i = 1, ..., n\} \qquad (2.2)$$

завъртането на ъгъл θ (докато главната ос съвпадне с тази на оста x) се задава със следното матрично уравнение:

$$\begin{pmatrix} x'_i \\ y'_i \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x_i \\ y_i \end{pmatrix}, i = 1, ..., n \qquad (2.3)$$

Траекторията от точки $S' = \{S'_i = (x'_i, y'_i), i = 1,...,n\}$ описва подписа след ротация.

На следващата фигура в ляво е показан собственият подпис от базата данни за потребител "Десислава Николова" и подписът след извършване на ротация:



Фигура 2.1 Извършване на ротация на подпис

Транслация на подпис в избрана начална точка

За даден подпис S с n на брой точки, представен във вида (2.2) и избрана начална точка с координати (x_o, y_o) първоначално се намират съответните компоненти на транслация T_x и T_y:

$$\begin{pmatrix} T_x \\ T_y \end{pmatrix} = \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} - \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} \qquad (2.4)$$

след което се извършва самата транслация, като за всяка двойка координати (x_i, y_i) от стойностите на x_i и y_i се извадят съответно стойностите T_x и T_y:

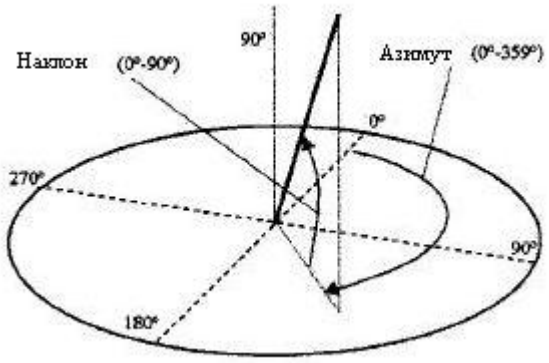
$$\begin{pmatrix} x_t^{Tr} \\ y_t^{Tr} \end{pmatrix} = \begin{pmatrix} x_i \\ y_i \end{pmatrix} - \begin{pmatrix} T_x \\ T_y \end{pmatrix}, i = 1,...,n \qquad (2.5)$$

В резултат на трансформацията, траекторията от точки $S^{Tr} = \{S_i^{Tr}(x_i^{Tr}, y_i^{Tr}), i = 1,...,n\}$ представлява подписа след транслация.

2.2 Извличане на признаци на он-лайн подпис от графичен таблет

С помощта на пакета за разработка на използвания *Tablet PC SDK 1.7* се получава динамична информация за събираните с него подписи. Тази информация е структурирана в пакети (по един за всяка точка от подписа) и съдържа множество от данни, което се поддържа от таблета. Някои от тези данни са следните: x и y координати, натиск, наклон на писалката, времеви отпечатък (*timestamp*) и азимут. Координатите определят местоположението на върха на писеца върху координатната система на таблета, натискът приема стойности в интервала [0,1023] и представлява приложения от писалката натиск в съответната точка. Наклонът на писалката е ъгъла, който тя сключва с повърхността на таблета и приема стойности в интервала [0,90], а

азимутът е ротацията по часовниковата стрелка на писалката в равнината XY и приема стойности в интервала [0,360] (Фигура 2.2).



Фигура 2.2 Наклон и азимут

Последователността от точки, изчертани без да се вдига писалката от повърхността на таблета се нарича щрих (*stroke*) и всеки подпис се състои от един или повече щриха. За всяка от точките на подписа се получава и информация за номера на щриха, към който принадлежи.

Стойностите на признаците за даден подпис се изчисляват въз основа на неговата динамичната информация. В Таблица 2.1 са представени използваните глобални признаци, като във формулите им са използвани следните означения. С *Si* е означена *i*-тата точка от даден подпис, с *n* – броя на точките в подписа, а (x_i, y_i) са координатите на точката *Si*. Натискът в точката *Si* означаваме с p_i ,с s_i – наклонът в *i*-тата точка и с t_i – изминалото време между изчертаването на *i*-тата и (*i*+*1*)-вата точка. С $(\bar{x}, \bar{y})$ сме означили координати на центъра, с (x_{lev}, y_{lev}) и с (x_{desen}, y_{desen}) - координатите на най-лявата и най-дясната точка на подписа.

Центърът $(\bar{x}, \bar{y})$ на подписа се изчислява по формулата:

$$(\bar{x}, \bar{y}) = (\frac{1}{n}\sum_{i=1}^{n} x_i, \frac{1}{n}\sum_{i=1}^{n} y_i). \qquad (2.6)$$

Разстоянието d между две точки от подписа (x_1, y_1) и (x_2, y_2) се задава с:

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}. \qquad (2.7)$$

Таблица 2.1 Глобални признаци

№	Признак	№	Признак
A1	Дължина на подписа *L*	A14	Разстояние от най-лявата точка до центъра
A2	Височина *H*	A15	Разстояние от центъра до най-дясната точка
A3	Сплеснатост *H/L*	A16	Ъгъл на отсечката, определена от центъра и най-лявата точка
A4	Брой точки *N*	A17	Ъгъл на отсечката, определена от центъра и най-дясната точка
A5	Общо време	A18	Разстояние от най-лявата точка до началната
A6	Брой сегменти	A19	Разстояние от най-дясната точка до крайната
A7	Плътност A4/A1*A2	A20	Ъгъл на отсечката, определена от най-лявата точка и началната
A8	Разстояние от началната точка до центъра	A21	Ъгъл на отсечката, определена от крайната и най-дясната точка
A9	Разстояние от крайната точка до центъра	A22	Брой щрихи
A10	Разстояние от началната до крайната точка	A23	Наклон на подписа
A11	Ъгъл на отсечката, определена от центъра и началната точка	A24	Средна стойност на натиск
A12	Ъгъл на отсечката, определена от центъра и крайната точка	A25	Средна стойност на наклон на писалката
A13	Ъгъл на отсечката, определена от началната и крайната точка	A26	Средна стойност на азимут

Ъгълът φ на отсечката от точка (x_1, y_1) до точка (x_2, y_2) от подписа се задава с:

$$\varphi = arctg\left(\frac{(y_2 - y_1)}{(x_2 - x_1)}\right). \qquad (2.8)$$

Една точка *Si* е локален екстремум, ако е изпълнено едно от условията (2.9):

$$\begin{aligned} y_{i-\varepsilon} < y_i > y_{i+\varepsilon} \\ y_{i-\varepsilon} > y_i < y_{i+\varepsilon} \\ x_{i-\varepsilon} < x_i > x_{i+\varepsilon} \\ x_{i-\varepsilon} > x_i < x_{i+\varepsilon} \end{aligned} \qquad (2.9)$$

Големината на ε се определя експериментално, като може да бъде равна на 2,3..., но рядко се използва $\varepsilon = 5$. Всяка двойка последователни екстремуми разделя подписа на сегменти.

За някои от горепосочените признаци нямаме данни да се срещат в литературата до момента. Те се изчисляват лесно и имат ясен физически смисъл, което ги прави разбираеми за експертите в областта на криминалистиката. Номерата им са A8-A21.

2.3 Минимизиране на броя на признаците

В контекста на задачата за разпознаване на подписи съществуват два подхода при избор на признаците, а именно общи признаци за всички участници и специфично подмножество от признаци за всеки участник, обосноваващо се на факта, че подписите на хората са уникални сами по себе си [11]. Тъй като подписът е строго индивидуален, едни признаци са характерни за един участник, а други – за друг. Затова е необходимо да се извлекат тъкмо тези индивидуални признаци, които най-добре характеризират подписа на всеки участник.

В този раздел ще представим метод за избор на регресионни променливи, базиран на критерият *Cp* на *Mallows* [38]. Подборът на подмножество от независими променливи (фактори), които да се включат в регресионния модел, така че той да дава най-надеждна прогноза, е важна задача в регресионния анализ. Този проблем се нарича избор на най-добро подмножество или избор на най-добро регресионно уравнение [10]. Съществуват два противоречиви критерия за избор на подмножество от независими променливи. От една страна, избраният модел би следвало да включва колкото се може повече променливи, което би гарантирало надеждна прогноза на модела. От друга страна,

съществуват няколко причини за намаляването на броя променливите в модела. Първо, при по-малък брой променливи се получава по-прост модел, с който се работи по-бързо. Второ, намаляването на броя на променливите често води до намаляване на взаимозависимостта между тях. И накрая, оценяването на зависимата променлива чрез голям брой променливи би могло да доведе до внасянето на допълнителен шум и влошаване на оценката. Затова при избора на регресионни променливи е необходимо да се вземат предвид възможните предимства и недостатъци от редуцирането на включените в модела променливи.

2.3.1 Нормиране на признаците

Тъй като всеки от признаците получава стойности в различни интервали се налага те да се нормират. Чрез извършването на нормировка стойностите на данните попадат в обща скала на отчитане и се премахват единиците за измерване. Сред известните техники за нормиране са стандартизацията (нормировка *z-score*) и нормировката *min-max* [14]. В първия случай, стойностите на признаците се трансформират в такива със средна стойност, равна на нула и стандартно отклонение, равно на единица. Новополучената стойност f_i^{zscore} за признак f_i се намира по формулата:

$$f_i^{zscore} = \frac{f_i - \mu_{f_i}}{\sigma_{f_i}}, i = 1,...,k \qquad (2.10)$$

където с μ_{f_i} и σ_{f_i} са означени съответно средната стойност и стандартното отклонение за *i*-тия признак, пресметнати за всички подписи от съответния потребител, изчислени по формулите:

$$\mu_{f_i} = \frac{1}{n}\sum_{j=1}^{n} f_{ij}$$
$$\sigma_{f_i} = \sqrt{\frac{1}{n-1}\sum_{j=1}^{n}(f_{ij} - \mu_{f_i})^2} \qquad (2.11)$$

Нормировката *min-max* в интервала $[c,d]$ се извършва за стойностите на всеки признак по формулата:

$$f_i^{\min\max} = \frac{f_i - \min_{f_i}}{\max_{f_i} - \min_{f_i}}(d-c) + c, i = 1,...,k \qquad (2.12)$$

където с min_{f_i} и max_{f_i} са означени съответно минимума и максимума за *i*-тия признак,

пресметнати по всички подписи от съответния потребител.

$$\min_{f_i} = \min_j (f_{ij}) \qquad (2.13)$$

$$\max_{f_i} = \max_j (f_{ij})$$

В задачите за разпознаване на образи нормировката най-често се извършва в интервала [0,1].

2.3.2 Премахване на един от група признаци с висока корелация

Корелацията е мярка за измерване както на посоката, така и на силата на връзката между две променливи [56]. Стойността на корелацията се определя от стойността на корелационния коефициент. Най-често използван е корелационният коефициент r на Pearson, който за два p-мерни вектора $\mathbf{X_1}$ и $\mathbf{X_2}$ се определя по формулата:

$$r_{X_1,X_2} = \frac{\sum_{i=1}^{p}(X_{1i} - \mu_{X_1})(X_{2i} - \mu_{X_2})}{\sqrt{\sum_{i=1}^{p}(X_{1i} - \mu_{X_1})^2 \sum_{i=1}^{p}(X_{2i} - \mu_{X_2})^2}} \qquad (2.14)$$

където с μ_{X_j} е означена средната стойност на вектора $\mathbf{X_j}$, $\mathrm{j} = 1,2$

$$\mu_{X_j} = \sum_{i=1}^{p} X_{ji}, j = 1,2 \qquad (2.15)$$

Големината на корелационния коефициент е показател за силата на линейната връзка между векторите $\mathbf{X_1}$ и $\mathbf{X_2}$. Колкото той е по-близко до 1 или -1, толкова е по-силна връзката между двата вектора. Стойността на коефициента на корелация би могла в голяма степен да се повлияе от един или повече признаци, силно отдалечени от групата данни (*outliers*) [56].

Корелационната матрица на признаците с размер $k \times k$ може да се представи по следния начин за даден потребител $m = 1,\ldots,M$

$$C^m_{k\times k} = \{C^m_{ij}\},\ i,j = 1,\ldots,k \qquad (2.16)$$

Всеки недиагонален елемент $C^m_{ij}, i \neq j$ на корелационната матрица представлява стойността на коефициента на корелация между признака i и признака j за съответния потребител m.

Методът на корелационните плеяди [99] се състои в намирането на групи от

признаци с висока стойност на корелация между признаците в групата и ниска стойност на корелация между признаците от отделните групи. Тези групи признаци се наричат корелационни плеяди. С прилагането на този метод се намалява броя на признаците като от всяка група се оставя единствен признак. Няма теоретически обосновано становище кой признак от плеядата да бъде запазен. Най-простият и практически оправдан подход е да се избере този признак, получаването на който е най-бързо и лесно. Значително по-убедително би било, ако за всяка комбинация от избрани признаци се оцени точността на класификацията. Това обаче изисква значителни изчислителни усилия, а резултатът от практическа гладна точка едва ли ще бъде съществен точно поради силната корелационна връзка в плеядата.

Надеждността на всеки един от корелационните коефициенти се определя от т.нар. ниво на значимост, което зависи от размера на извадката (в разглежданата задача това е броя на собствените подписи от дадения потребител) [56]. Например за $n = 10$ и доверителна вероятност, равна на 99%, съответното ниво на значимост е 0.7155 [98].

2.3.3 Избор на най-информативни признаци

Регресионният анализ е често използван метод за установяване на връзка между независимите променливи и зависимата променлива. Намира широко приложение в различни приложни задачи [10]. Изборът на регресионни променливи се състои в определянето на подмножество от променливи, които да се включат в модела, така че той да дава надеждна прогноза за принадлежността на дадена точка към коректния клас. Формулировката на задачата за избор на регресионни променливи е следната [54]. Нека са дадени $n \geq k+1$ наблюдения над k -мерния вектор от независими променливи $\mathbf{x}_\mathbf{j}^t = (x_{1j},...,x_{kj})$ и зависимата променлива y_j, $j = 1,...,n$, стойността на която се определя от:

$$y_j = \sum_{i=1}^{k} \beta_i x_{ij} + e_j \qquad (2.17)$$

Моделът на регресията (10) има следния матричен вид:

$$\mathbf{Y} = \mathbf{X\beta} + \mathbf{e} \qquad (2.18)$$

където $\mathbf{Y}$ е n -мерен вектор на зависимите променливи, $\mathbf{X}$ е матрица на независимите променливи с размерност $n \times k$, $\boldsymbol{\beta}$ е k -мерен вектор на неизвестните регресионни

коефициенти и **e** е n-мерен вектор на остатъците, за които се предполага, че са независими и нормално разпределени $N(0,\sigma^2)$ с математическо очакване, равно на нула и дисперсия, равна на σ^2. Регресионните параметри (коефициенти) β_i, $i=1,\ldots,k$ са неизвестни константи, които следва да се оценят по метода на най-малките квадрати от изходните данни. Техните оценки се означават с b_i. В модела може да се включи или изключи свободния член β_0 според решението на изследователя. В проведените експерименти (Глава 4) свободният член е премахнат от уравнението на регресията. Тогава, векторът **b** от оценките на вектора на коефициентите по метода на най-малките квадрати се задава чрез:

$$\mathbf{b} = (\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{Y} \qquad (2.19)$$

Не съществува универсален метод за избор на най-добър регресионен модел. За сравняване на различните регресионни модели са необходими критерии. Такива са например критерия *Cp*, различните информационни критерии *AIC* (*Akaike information criteria*), *BIC* (*Bayesian information criterion*) и др. [10]. Критерият *AIC* представлява мярка за качеството на даден статистически модел, построен върху дадено множество от данни и затова може да служи като средство за избор на модел. Основава се на ентропията на информацията като дава относителна оценка на количеството загубена информация при използване на даден модел. Свързан с този критерий е и критерия *BIC*, който е базиран на функцията на правдоподобие (*likelihood function*) в статистиката и който също представлява начин за избор на модел от краен брой модели. Добавянето на параметри в модела може да увеличи степента на правдоподобие, но това би могло да доведе до пренастройване на модела. За решаване на този проблем, при прилагането на критериите *AIC* и *BIC* се въвежда понятието *penalty term* за броя параметри на модела.

Важно е да се отбележи, че не съществува единствено, уникално най-добро множество от променливи, а съществуват няколко адекватни множества от променливи, които да се използват за формиране на модела. Различните множества от подходящи и адекватни променливи разкриват структурата на данните и помагат за по-добро вникване в информацията, която носят. Процесът по избор на променливи би могъл да се разглежда като анализ на корелацията между факторите и начина, по който те самостоятелно и в тандем оказват влияние върху зависимата променлива. Най-популярният подход за избор на регресионен модел е този за "избор на най-добро подмножество" (*best subset*). При прилагането му се разглеждат всичките $2^k - 1$ на брой

подмножества от регресионни променливи. Това на практика е неприложимо, когато броят на променливите е голям. В литературата съществуват методи, които позволяват да се намери най-добро или почти най-добро подмножество от променливи, без да е необходимо да се извършват пресмятания върху всички подмножества от признаци, а само върху част от тях [55]. Към тях се отнася и разглеждания по-долу метод на *Hocking* и *Leslie*. Съществуват и методи за избор на регресионни променливи, при които не се изисква изчисляването на всички възможни уравнения и при тях на изследователя не се предоставя голям обем информация. Те са свързани със значително по-малко изчисления и в някои случаи се явяват единствения възможен начин за решение на задачата [10]. За тях е присъщо, че на всеки етап се включва или изключва една променлива. Наричат се "отгоре-надолу" (*backward elimination*) и "отдолу-нагоре" (*forward selection*) и се основават на неявното допускане за съществуването на единствено най-добро уравнение, докато често на практика съществуват няколко еднакво добри уравнения, съдържащи различен брой променливи. При първия подход се започва с пълен регресионен модел и последователно на всяка стъпка се премахва най-малко значимата променлива според даден критерий, докато при втория една по една се добавят независимите променливи като на всяка стъпка към уравнението се добавя променливата с най-голяма редукция на регресионната сума от квадрати. При метода "отдолу-нагоре" последователно за всяка от независимите променливи се изчислява F статистиката, която дава информация за влиянието ѝ върху модела, ако съответната променлива участва в него. Изчислените p-стойности за тези F статистики по нататък се сравняват с предварително дефинирана стойност за модела (по подразбиране равна на 0.50). Ако няма F статистика с ниво на значимост, по-голямо от предварително дефинирана стойност за модела, процедурата спира. В противен случай, към модела се добавя променливата с най-голяма стойност на F статистиката. Веднъж добавена променлива към модела, остава в него. По нататък се изчисляват F статистиките за променливите извън модела и се извършват горепосочените оценявания. Процедурата продължава с добавянето на променливи към модела, докато не остане променлива със значима стойност на F. При метода "отгоре-надолу" пък се започва с изчисляването на F статистиките за модела, съдържащ всичките независими променливи. След това променливите се премахват от модела една по една, докато не се постигне модел, съдържащ променливи със значими стойности на F статистиките според предварително зададено ниво на значимост на модела. Изобщо, на всяка стъпка се премахва тази независима променлива, която демонстрира най-малко влияние върху

модела. Освен тези два основни подхода, съществуват и техни модификации [10] като например стъпковия метод, който представлява вариант на метода "отдолу-нагоре" с тази разлика, че веднъж добавена променлива към модела невинаги остава в него. Следва да се отбележи, че и при тези две основни стратегии за избор на регресионни променливи и техните модификации, процедурата се прилага автоматично, докато при прилагането на подхода "избор на най-добро подмножество" е необходимо и участието на изследователя при вземането на окончателното решение за избор.

В дисертационната работа е разгледан и приложен метод, който представлява по същество модификация на подхода "избор на най-добро подмножество" за определяне на участващите в регресионния модел независими променливи. Повечето от критериите за избор на регресионни променливи са монотонни функции на остатъчната сума от квадрати (RSS, residual sum of squares) и поради тази причина задачата за избор на регресионни променливи се редуцира до намирането на тези подмножества от променливи, за които остатъчната сума от квадрати е малка. Остатъчната сума от квадрати $\boldsymbol{RSS_p}$ за n наблюдения над p на брой променливи се дефинира като:

$$RSS_p = \sum_{j=1}^{n}(y_j - \sum_{i=1}^{p}\beta_i x_{ij})^2 \qquad (2.20)$$

Mallows [57] предлага критерий за сравняване на няколко регресионни модела с цел откриването на най-добрите от тях. За целта се пресмята стойността на за всеки един от моделите. За модел с *p* на брой променливи и *n* наблюдения, се дефинира като:

$$C_p = \frac{RSS_p}{\hat{\sigma}^2} - (n - 2p) \qquad (2.21)$$

Оценката $\hat{\sigma}^2$ на σ^2 се задава по формулата:

$$\hat{\sigma}^2 = \frac{1}{n-k}\sum_{j=1}^{n}(y_j - \sum_{i=1}^{k}\beta_i x_{ij})^2 = \frac{RSS_k}{n-k}. \qquad (2.22)$$

където с σ^2 е означена дисперсията на *n*-мерния вектор на остатъците $\mathbf{e}$:

$$\mathbf{e} = \mathbf{y} - \sum_{i=1}^{k}\beta_i x_i \qquad (2.23)$$

Така, ако даден модел с p на брой променливи е добър, то от (2.21) и (2.22) се намират следните стойности за математическото очакване на стойностите на RSS_{p} и C_p:

$$E(RSS_p) = (n-p)\hat{\sigma}^2 \qquad (2.24)$$

$$E(C_p) \approx p$$

Следователно регресионно уравнение с малко отклонение има очаквана стойност на C_p, близка до тази на p . Графиката на C_p и p показва подходящите модели като точки близки до правата с уравнение $C_p = p$. За добри подмножества от променливи се считат тези с малки стойности на C_p или със стойности на C_p, близки до p [57].

Hocking и *Leslie* [55] създават метод, който позволява откриването на най-добри подмножества от регресионни променливи след разглеждането на малка част от всички $\binom{k}{p}$ възможни подмножества с размер p. *LaMotte* и *Hocking* [29] модифицират метода така, че сравнително големи по обем задачи могат да се решават с минимум изчисления. С т.нар. r-подмножество се означава подмножеството от променливи, които се изключват от регресионния модел, а с p-подмножество – подмножеството от променливи, които остават в модела. По същество методът се базира на т.нар. редукции на m-подмножества, т.е. на редукции в остатъчните суми от квадрати поради премахването на подмножества с размер m от модела с k на брой променливи. При оригиналния метод $m = 1$, докато при модифицирания метод $1 \le m \le 4$. Редукциите на m-подмножества се използват за определянето на най-добро r-подмножество, което да се премахне (за $r > m$). Редукцията на регресионната сума от квадрати Red_r, получена от премахването на подмножество от r променливи се задава чрез:

$$Red_r = RSS_p - RSS_k \qquad (2.25)$$

Множеството от r променливи, за което стойността на тази редукция е най-малка определя подмножеството от p, $p = k - r$ регресионни променливи, за което остатъчната сума от квадрати е минимална. В [55] е предложена следната формула за намиране на C_p чрез тази редукция:

$$C_p = \frac{Red_r}{\hat{\sigma}^2} - (2p - k) \qquad (2.26)$$

По-долу са изброени стъпките на обобщения алгоритъм [29].

1. В началото се намират регресионните коефициенти за модела с всичките k на брой променливи;

2. За всяка от променливите x_i се пресмята редукцията θ_i от премахването й (т.нар. редукция на една променлива, *univariate reduction*). Редукцията от премахването на *i*-тата променлива се задава с:

$$\theta_i = \hat{\sigma}^2 t_i^2 \qquad (2.27)$$

където t_i^2 (квадратът на *t*-статистиката на *i*-тия регресионен коефициент) се пресмята по формулата:

$$t_i^2 = \frac{b_i^2}{\hat{\sigma}_{b_i}^2} \qquad (2.28)$$

Стандартната грешка на оценения коефициент b_i се означава с $\hat{\sigma}_{b_i}^2$ и се изчислява като корен квадратен от диагоналния елемент на позиция (i,i) от ковариационната матрица на коефициентите.

Процедурата продължава с пренареждане на променливите по нарастване на съответните им редукции:

$$\theta_1 \leq \theta_2 \leq \cdots \leq \theta_k$$

Методът се основава на следното фундаментално свойство на квадратичните форми: "Ако редукцията на регресионната сума от квадрати, получена от премахването на подмножество от променливи с най-голям индекс j не надвишава стойността на редукцията θ_{j+1} от премахването на $(j+1)$-вата променлива, то не съществува подмножество, включващо променливи с индекси, по-големи от j, за което редукцията да е по-малка".

3. Пресмятат се редукциите на m-подмножества за всяко от всичките $\binom{k}{m}$ на брой подмножества с размер m. Всяко от тези подмножества с размер m се представя във вида $(i_1, i_2, ..., i_m)$, където $i_1, i_2, ..., i_m$ са променливите, от които е съставено,

подредени в нарастващ ред, $1 < i_j < k$ и $i_1 < i_2 < ... < i_m$. Така пресметнатите редукции на *m*-подмножества се подреждат в нарастващ ред. Тези *m*-подмножества, чиито първи индекс е по-голям или равен на $(r - m + 1)$ служат за дефиниране на етапи за изследване на r - подмножества. Подмножествата с размер *r*, определени на даден етап съдържат *m* на брой индекси от съответното m-подмножество и $(r - m)$ индекси, по-малки от първия индекс в съответното m - подмножество. Броят на дефинираните по този начин етапи е $\begin{pmatrix} k-r+m \\ m \end{pmatrix}$. Етапите се номерират според големината на стойността на редукцията на m-подмножествата, които ги дефинират. Така, първият етап се състои от r-подмножествата, определени от m-подмножеството с най-малка стойност на редукция.

4. Изобщо, на всеки етап *q* се прави проверка дали най-малката от редукциите на r -подмножествата, дефинирани от всички етапи до момента $(1,...,q)$ е по-малка от редукцията на m-подмножеството, дефиниращо следващия (*q+1)-ви* етап, и ако това е изпълнено, се счита, че е открито търсеното *r*-подмножество, което да напусне изходния модел и съответстващото му най-добро *p*-подмножество, а в случай, че не е изпълнено, се преминава към следващия (*q+1)-ви* етап.

По-такъв начин се решава задачата за намиране на най-добро подмножество от променливи, които да останат в регресионния модел.

2.4 Резултати в Глава 2

1. Формулирани са изискванията към използваните за верификация признаци.
2. Описани са трансформациите на получените от графичния таблет изображения на подписи, необходими за привеждането им в стандартен вид и извличането на признаци. Представени са 26 глобални признака, необходими за математическото описание на подписа. Признаците с номера A8-A21 не се срещат в литературата до момента.
3. Описан е подход, основаващ се на последователното прилагане на метода на корелационните плеяди и C_p статистиката на *Mallows*, за избор на най-информативно от гледна точка на класификацията индивидуално за всеки потребител подмножество от признаци.

Глава 3. Класификация

Класификацията се състои в намирането на съответствие между пространството от признаци и пространството от етикети на класове или с други думи – в отнасянето на даден образец към съответния му клас въз основа на стойностите на неговите конкретни признаци. Разграничават се следните два типа класификация: дихотомия (класификация в два класа) и мултикласова (класификация в повече от два класа). Следва да се отбележи, че не съществува "най-добър" класификатор, понеже класификаторите, приложени за решаването на различни задачи и обучени чрез различни алгоритми, се представят различно. В областта на разпознаването на образи е известна т.нар. *No Free Lunch Theorem*, според която не съществува идеално решение на която и да е класификационна задача [67] (теоритично Бейсовият класификатор е опттимален, но за построяването му е необходимо предварително познаване на вероятностните плътности на класовете). С други думи, няма основание да се твърди, че даден класификатор ще се представи най-добре върху която и да е задача. При избора на класификатор, трябва да се вземат предвид типа на решаваната задача, броя на признаците, броя примери за обучение и др., които имат отношение към определянето му. В случай, че даден класификатор се представя по-добре от друг в конкретна ситуация (с конкретни данни), това не означава, че той изобщо е по-добър от него при решаването на друга задача.

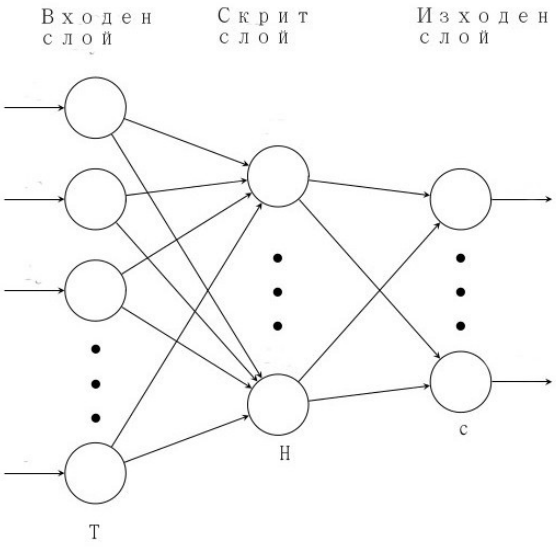
Тази глава от изследването е посветена на построяването на класификатори и оценяването на тяхното качество. По-конкретно са разгледани изкуствените невронни мрежи и класификатора по *k*-най-близки съседи, като класификатори при решаване на задачата за разпознаване по подпис.

Идеята за изкуствени невронни мрежи произлиза от идеята за създаване на математически модел на човешките интелектуални способности. В крайна сметка невронните мрежи се превръщат в надеждно средство за класификация в областта на разпознаването на образи. Архитектурата (топологията) на всяка невронна мрежа се състои от няколко обработващи елементи, наричани (изкуствени) неврони [34]. Те са организирани в слоеве и са свързани помежду си с връзки с определени теглови коефициенти (тегла на връзките). Всяка невронна мрежа се състои от един входен, няколко скрити (един или повече) слоеве и един изходен слой. Всеки изкуствен неврон преобразува входните сигнали, които получава от сензори от околната среда или от други неврони и формира съответен изходен сигнал, който разпространява към невроните, с които е свързан. За получаването на изходния сигнал, първо входните сигнали се умножават по съответните тегла, които определят силата на връзките и

резултатите се събират като по този начин се формира линейния суматор (*net sum*) за дадения невро*н, след което* този линеен суматор се трансформира в изходния сигнал на неврона чрез специфичната за мрежата активационна функция (*activation function*). Изборът на активационна функция е от голямо значение за работата на невронната мрежа и за нейното обучение. Съществуват множество активационни функции, всяка от които води до различно поведение на невронната мрежа. Най-често използваните функции са сигмоидална (логистична) функция, функцията идентитет, праговата функция [34].

Задачата, която решават невронните мрежи може да се разглежда като задача за намирането на съответствие $f : I \to O$ между дадено входно множество I и изходно множество O. Невронната мрежа представя функция $y = f(x)$, където $y \in O$ и $x \in I$. Тъй като самата класификация представлява намирането на съответствие между пространството от признаци и множеството на класовете, то невронните мрежи могат да бъдат използвани като класификатори [76].

На следната фигура е представена права мрежа с T на брой неврони (равен на броя на признаците) във входния слой, H на брой неврони в скрития слой и c на брой неврони в изходния слой (равен на броя на класовете).



Фигура 3.1 Архитектура на права мрежа

Преди да се извърши избор на подходяща архитектура на невронната мрежа, се извършват множество експерименти с различни стойности на параметрите, на

началните стойности на теглата и т.н. [34]. Веднъж обучена, дадена невронна мрежа може да бъде много бърза при разпознаването на новопостъпил образец. Недостатък на невронните мрежи е, че са нестабилни поради високата им чувствителност към началните условия и че често обучението им е бавно и със слаба сходимост. Поради тези причини, прилагането на невронни мрежи за решаването на конкретна задача на практика не е тривиално.

При традиционния подход за решаване на задачата за разпознаване на образи се проектират L на брой класификатори $D_1,....,D_L$ (или L на брой версии на единствен класификатор) по отношение на множество от данни за обучение и се оценяват техните грешки чрез тестването им върху валидираща извадка. За най-добър се избира този от класификаторите, който води до най-малка грешка при разпознаването. Този подход работи сравнително добре при наличието на голямо и представително множество от данни [14]. Обаче на практика обикновено на разположение са ограничени по обем данни и поради това оцененото качество може съществено да се различава от качеството на класификатора, което би се постигнало на практика и разбира се, това прави избора на най-добър класификатор невъзможен като дори съществува възможността от избор на най-неподходящия от класификаторите. Интуитивно е ясно, че “по-безопасно” е вместо да се избира един класификатор да се използват всичките налични L класификатори като се осреднят изходите им [1], защото по този начин ще се намали риска от избирането на класификатор с ниска точност. (По този начин обаче със сигурност ще бъде изпуснат и най-добрия класификатор. Този подход се характеризира с минимизация на средните загуби). И така, алтернативен на традиционния подход е подхода, при който наведнъж и съвместно се използват няколко класификатора или създаването на т.нар. мултикласификатор. Оказва се, че едновременното комбиниране на различни класификатори подобрява значително точността на класификацията. При провеждането на редица научни изследвания се забелязва, че въпреки че един от класификаторите в традиционния подход има най-добро представяне, множеството от неправилно класифицирани образци от различните класификатори не се припокриват, т.е. различните класификатори правят грешки на различни образци. Оттук идва и допускането, че различните класификатори предлагат допълваща се информация и че чрез комбинирането на техните индивидуални мнения при вземането на окончателното решение би могло да се подобри качеството на класификацията. Доказано е, че с подходящо комбиниране на резултатите, получени от различните експерти по някакво

правило може да се получат резултати, по-добри от тези на отделните експерти [46]. В идеалния случай при комбинирането се отчита силата и точността на отделните класификатори, като се избягват техните слабости.

Известно е, че невронните мрежи често намират приложение като базови класификатори в мултикласификатор [46]. Тъй като даден мултикласификатор може да представлява комбинацията от един и същ по вид класификатор, но с различни стойности на параметрите, то невронните мрежи биха могли да бъдат базови класификатори в него [46]. Използването на мултикласификатор от невронни мрежи би могло да опрости проблема за избор на адекватни стойности на някои параметри (напр. броя на неврони в скрития слой), а също и да бъде възможно решение за редуциране на грешката при обобщаване (*generalization error*) на невронната мрежа [72]. Комбинирането на невронни мрежи, получени за различни стойности на началните параметри (в случая началните тегла), редуцира риска от избирането на един класификатор от тип невронна мрежа, при който обучаващият алгоритъм попада в локален минимум.

3.1 Невронните мрежи като класификатори

Изкуствените невронни мрежи (или само невронни мрежи) възникват като реализация на идеята за създаване на математически модел на невралната структура на човешкия мозък, който да отразява неговата способност за обучение, основано на опита. Невронните мрежи биха могли най-общо да бъдат обучени да решават следните видове задачи: апроксимация на функции и класификация на образи. Те могат да изпълняват сложни задачи и да решават проблеми, трудни за конвенционалните компютри и за човека в различни области от разпознаването на образи [34], сред които разпознаване на подписи [20], говор [81], лица [82] и др. Успешната реализация на проекти с помощта на невронни мрежи за класификация [50] в различни практически области като разпознаване на образи, идентификация, класификация, системи за контрол и др. се дължи най-вече на способността им за обобщаване и обучаване чрез примери. Значимото им предимство се изразява в това, че те могат да бъдат разглеждани като самоадаптивни методи, базирани на данните, като универсални апроксиматори на функции, а също и като нелинейни модели за моделиране на сложни процеси и функции [34, 83]. Възможността за универсална апроксимация на функции е доказана за двата важни модела на невронни мрежи – многослойния перцептрон [80] и невронните мрежи с радиални функции [84]. Същевременно те са лесни за употреба,

тъй като се обучават чрез примери, а задачата на изследователя се състои в осигуряването на достатъчно представително множество от данни на невронната мрежа, която чрез подходящ обучаващ алгоритъм да открие закономерностите в данните.

3.1.1 Видове невронни мрежи

Архитектурата на невронната мрежа се определя от начина, по който са свързани нейните неврони. Най-често срещани са невронните мрежи, съставени от няколко (последователни) слоя от елементи, при които елементите от най-ниския слой играят ролята на входни устройства на мрежата (те получават информация от външната среда), а елементите от изходния слой играят ролята на изходни устройства на мрежата (те извеждат резултата от работата на мрежата, който се получава на базата на входните сигнали и теглата на връзките между елементите). Този тип невронни мрежи се наричат прави мрежи с право разпространение (*feedforward networks*). При тях връзките са еднопосочни и свързват елементите от един слой с елементи на слоя, разположен непосредствено над него, тоест информацията се разпространява само в една посока напред – от входа към изхода. В зависимост от броя на слоевете в мрежата се различават *еднослойни* невронни мрежи (само с един входен и един изходен слой, при които липсват т. нар. *вътрешни* или *скрити слоеве*) и *многослойни* невронни мрежи (с поне един скрит слой). Скритите (междинни) слоеве служат само за обработка. Другият тип невронни мрежи се наричат рекурентни (невронни мрежи с обратна връзка) , при които всеки елемент е свързан с двупосочни връзки с всички свои съседи. Те се различават от правите невронни мрежи по това, че имат поне една обратна връзка.

3.1.2 Обучение на невронна мрежа

За обучението на дадена невронна мрежа за решаването на конкретна практическа задача е нужно наличието на достатъчен брой примери за обучение на вярно поведение на мрежата, състоящи се от съответните им входни и изходни вектори. Самото обучение се състои в итеративно актуализиране на стойностите на теглата на мрежата с цел оптимизиране на качеството ѝ. Правилото, по което се променят теглата на връзките, се нарича обучаващо правило (обучаващ алгоритъм) на мрежата. Целта е тя да бъде в състояние да асоциира коректно примерен вход към желан изход. Най-често прилаганата функция за оценка на качеството за прави невронни мрежи е средно квадратичната грешка *mse* (*mean square error*) (3.1) на получения резултат y относно целевия резултат t:

$$mse = \frac{1}{m}\sum_{i=1}^{m}(e_i)^2 = \frac{1}{m}\sum_{i=1}^{m}(t_i - y_i)^2 \qquad (3.1)$$

Най-често прилаганият обучаващ алгоритъм е алгоритъмът с обратно разпространение на грешката (*backpropagation algorithm*). Този алгоритъм е много популярен, тъй като е концептуално прост, изчислително ефикасен и обикновено дава задоволителни резултати. Обучението чрез този алгоритъм включва три етапа: разпространение напред на обучаващия образец, пресмятане на съответната грешка, нейното разпространение назад и актуализиране на стойностите на теглата. Той е познат в литературата и като алгоритъм за спускане по градиента (*gradient descent algorithm*). При него стойностите на теглата и отклоненията се променят в посоката на най-бързото намаляване на функцията за оценка на качеството на мрежата. Този обучаващ алгоритъм е много мощен, тъй като при наличието на достатъчен брой скрити неврони с него може да се обучи всяка функция. Обаче при наличието на прекалено голям брой скрити неврони се стига до състояние, в което мрежата има склонност да запаметява начина, по който да отговаря на входните стойности, а не да обобщава (т.нар. пренастройване на мрежата, *overfitting*).

В литературата [34] могат да се открият множество вариации на алгоритъма, предложени с цел подобряване на скоростта на обучение на мрежата, а също и други обучаващи правила като: *Gradient descent with momentum*, *Variable learning rate*, *Conjugate gradient algorithms, Quasi-Newton method*, *Levenberg-Marquardt algorithm.* В Таблица 3.1 са описани част от най-разпространените алгоритми със съответните им наименования и характеристики в инструментариума Neural Network Toolbox™ [47] на MATLAB.

За целите на обучението на невронната мрежа, множеството от образци (с размерност *N*) се разделя в две извадки: за проектиране (с размерност *Ndes*) и за тестване (с размерност *Ntst*) като *N* = *Ndes* + *Ntst*. Образците от извадката за проектиране се използват многократно за определяне на мрежовата архитектура, обучаващия алгоритъм и неговите параметри. Образците от тестовата извадка се използват еднократно за получаването на оценка на качеството на НМ върху непознати и неизползвани за нейното обучение данни. Те не се използват по време на обучението, а за сравняването на различни мрежови архитектури и за проверка на възможността на мрежата за обобщаване. Способността за обобщаване е изключително желана и критична характеристика на невронните мрежи, тъй като основната им цел е именно

извършването на точни прогнози върху нови данни.

Таблица 3.1 Алгоритми за обучение на НМ в MATLAB

Команда	**Наименование**	**Характеристики**
traingd	Gradient Descent	Оригинален, но най-бавен
traingdm	Gradient Descent с моментум	По-бърз от traingd
traingda	Gradient Descent с адаптивен α	По-бърз от traingd, само за обучение в пакетен режим
traingdx	Gradient Descent с адаптивен α и моментум	По-бърз от traingd, само за обучение в пакетен режим
trainrp	Resilient Backpropagation	Бърза сходимост
trainscg	Scaled Conjugate Gradient	Бърза сходимост
trainbfg	Алгоритъм BFGS	Quasi-Newton алгоритъм, бърза сходимост
trainoss	Алгоритъм One Step Secant	Quasi-Newton алгоритъм, бърза сходимост
trainlm	Levenberg-Marquardt	По-бързо обучение, редукция на използваната памет
trainbr	Бейсова регуляризация	Подобрява способността за обобщаване

От съществено значение за обучението на всяка невронна мрежа е това в кой момент да спре нейното обучение. В идеалния случай процесът по обучение е под прякото наблюдение на изследователя и обучението спира тогава, когато качеството започне да се влошава. Но това често е неприложимо на практика и затова е необходимо да се приложи някакво стопиращо условие (*stopping condition*), с което да се гарантира, че времето за обучение е крайно и че мрежата не е нито преобучена/пренастроена (*overtrained/overfitted*), нито недостатъчно обучена (*undertrained*). Дадена невронна мрежа е недостатъчно обучена, когато обучаващата функция не се настройва достатъчно добре към данните за обучение, което води до значителни грешки при класификацията. В случай на преобучение, невронната мрежа апроксимира примерите за обучение много добре, тъй като е запомнила как да отговаря на тях, но не обобщава правилно, когато се използва за класификация на непознати данни. С други думи, в този случай ефективността на мрежата върху примерите за обучение продължава да нараства, докато качеството върху непознатите данни се влошава. Причина за това би могло да бъде недостатъчния брой данни за обучение с наличие на шум или прекалено

сложния модел на невронната мрежа. Преобучение възниква, когато грешката при тестване нараства при същевременно намаляване на грешката при обучение. Най-добър апроксимиращ модел е този, получен за минимум на грешката при валидация при приемлива стойност на грешката при обучение.

Намаляването на ефекта от преобучение на мрежата би могло да се постигне чрез намаляване на сложността на мрежата, оптимизиране на нейната топология или по-ранно спиране на процеса на обучение (*early stopping*). По-ранното спиране на процеса на обучение е често прилагана техника за постигане на по-добра способност за обобщаване на мрежата. Прилагането на стопиращото условие води до прекратяване на обучението, ако са изпълнени определени критерии. Най-простото стопиращо условие е да се зададе максимален брой епохи, при достигането на който обучението да се преустанови. Но тъй като в повечето проблемни области (вкл. и в разпознаването по подпис) е налице известна вариация в обучаващите примери, то броят на епохите, необходими за задоволително обучение може да варира в широки граници. Използването на такъв вид стопиращо условие често води до преобучение на невронната мрежа или до недостатъчното й обучение. Друг вид стопиращо условие е чрез използването на целево ниво на грешката (*target error rate*), при което обучението спира, когато грешката при обучението падне до тази стойност. Но тъй като няма гаранция, че грешката при обучение ще спадне до желаната стойност, то е трудно да се предположи максималната стойност на грешката. Друго, по-удачно стопиращо условие е базирано на т.нар. *минимално ниво на подобрение*. Ако грешката при класификация не се подобри поне с някаква предварително зададена стойност за даден брой епохи (*размер на прозореца*), тогава обучението спира. Това означава, че невронната мрежа продължава обучението си докато съществува приемливо основателно подобрение и спира при достигане на плато.

След приключване на обучението на дадена невронна мрежа, тя може да бъде използвана за разпознаването на данни, неизползвани при обучението. Това става като на входа й се подадат тестови образци с техните характеристики и след това се изчисли резултата на изхода й, който служи за вземането на решение за разпознаване. Дори и обучението да е протекло бавно, веднъж обучената невронната мрежа може да даде отговор за даден тестов образец много бързо.

3.1.3 Предимства и недостатъци на невронните мрежи

Основното предимство на невронните мрежи е способността им за обучение и обобщаване. Обучението се състои в това те сами да откриват закономерностите в данни за обучение, а обобщаването – да предсказват точно стойността на изходния сигнал дори и за входни сигнали, неизползвани при обучението. Наред с това те са гъвкави, понеже биха могли лесно да се обучат отново, ако това се налага и са адаптируеми към постоянно променящата се информация, постъпваща отвън. Това, както и изчислителната им мощ, дължаща се на извършваните от тях паралелни изчисления, ги превръща в мощен инструмент за формулиране и проектиране на нелинейни модели за голям брой реални приложения (напр. поставяне на медицински диагнози, разпознаване по почерк и др.) дори и за данни с голяма размерност [34, 50]. В контекста на задачата за разпознаване, невронните мрежи не само определят принадлежността на образец към даден клас, но и дават информация за точността на техният отговор чрез т.нар. ниво на достоверност (*confidence level*). На разположение на изследователите са голям брой мрежови архитектури, с които да експериментират. Въпреки че изчисленията, обичайно извършвани от невронните мрежи са с високи изисквания по отношение на изчислителни ресурси, те са оптимизирани до такава степен, че биха могли да се извършват от персонални компютри за разумен интервал от време.

Като недостатък би могло да се изтъкне необходимостта от провеждането на голям брой експерименти до достигането на задоволителен модел, дългото време, необходимо за обучението на невронна мрежа за голямо множество данни за обучение, дори и при извършването на паралелни изчисления, а също и липсата на прозрачност при извършването на изчисленията от невронната мрежа, поради което не е възможно да се интерпретира построения модел. Също така, невронните мрежи са нестабилни, тъй като малки промени в данните за обучение могат да доведат до големи промени в самия класификатор, както в неговата структура, така и в неговите параметри [50].

3.1.4 Невронна мрежа за класификация

За даден входен образец $X = (x_1, ..., x_T)$ и множество от класове $\Omega = \{\omega_1, ..., \omega_c\}$, всеки изходен неврон ω_i оценява вероятността y_i за принадлежност към класа *i* чрез:

$$y_i = f\left\{\sum_{k=1}^{H} \omega_{ik}^{om} f\left\{\sum_{j=1}^{T} \omega_{kj}^{mi} x_j\right\}\right\} \qquad (3.2)$$

където ω_{kj}^{mi} е теглото на връзката между *j*-тия входен неврон и *k*-тия скрит неврон, ω_{ik}^{om} е теглото на връзката между *k*-тия скрит неврон и изхода, съответен на *i*-тия клас, а активационната функция е сигмоидалната функция $f(x) = 1/(1+e^{-x})$. Невронът с максимална стойност се избира за победител, а съответния му клас за резултат от класификацията. В същото време, изходите от невронната мрежа представляват оценки на Бейсовите апостериорни вероятности, ако активационната функция приема стойности в [0,1] и сумата от изходите е равна на единица (каквато е и посочената по-горе сигмоидална функция) [94]. Ако невронната мрежа е обучена да минимизира стойността на квадратичната грешка между изхода на мрежата и целевата стойност, то всеки от изходите y_i представлява апроксимация на апостериорната вероятност $P(\omega_i|x), i = 1,...,c$. В [95] са изложени следните предположения за модела на невронни мрежи, които на практика биха могли да не бъдат изпълнени: 1) че мрежата е достатъчно сложна, за да може да моделира апостериорните вероятности; 2) че на разположение са достатъчно на брой данни за обучение; 3) че оптимизационния алгоритъм може да намери глобален минимум на функцията на грешката.

3.1.5 Приложение на невронните мрежи в задачата за разпознаването по подпис

Невронните мрежи са подходящо средство за решаване на сложни задачи като разпознаването на подпис поради следните причини [51]. На първо място, в случай че са добре обучени, те обикновено дават верни отговори за входни образци, неизползвани при обучението. Също така многослойните невронни мрежи могат да моделират всяка функция на множество от променливи, а подписите могат лесно да се представят като такава функция. Поради възможността им за обобщаване, невронните мрежи могат да се справят с разнообразието и изменчивостта, характерни за образците на подписите. Тъй като в повечето случаи обучението на невронна мрежа отнема дълго време, в приложенията за разпознаване по подпис то обикновено се извършва еднократно и в оф-лайн режим, като по този начин на потребителите се спестява неудобството да чакат до приключването му. През последните години в литературата са предложени няколко системи за разпознаване на подпис, създадени чрез прилагането на невронни мрежи [51]. Идеалното обучение на една невронна мрежа за разпознаване

по подпис е практически невъзможно, поради нестабилния характер на данните (подписът на даден потребител зависи от възрастта му, от неговото физическо и психическо състояние, от употребата на медикаменти и опиати, от вида на хартията, върху която се подписва, от вида на писалката и още много други).

Съществуват два режима за реализация на система за разпознаване на подписи с невронни мрежи, а именно верификация и идентификация. За режим верификация са необходими както собствени, така и подправени подписи от всеки потребител. За всеки участник се създава и съхранява единствена невронна мрежа, която има само един изходен неврон, чийто краен отговор е ниво на достоверност, приемащо стойности между нула (подправен подпис) и единица (собствен подпис), която показва степента на сходство на подписа за тестване с подписите, използвани при обучението за съответния участник. Тази стойност се сравнява с предварително избрана стойност на прага за участника и ако я надвишава, подписа за тестване се счита за собствен, а в противен случай за чужд. Този подход е широко приложим на практика и позволява бързото добавяне и премахване на подписи на нови и стари потребители. Така, при добавяне на нов потребител е необходимо единствено да се обучи малка по размер невронна мрежа. В реалните приложения за разпознаване по подпис не се разполага с фалшифицирани подписи за обучение, а без наличието на отрицателни образци, всяка невронна мрежа може лесно да се преобучи. Ако пък в системата за разпознаване по подпис присъстват и фалшифицирани подписи за обучение, то оценка на класификацията (избрани признаци и класификационния алгоритъм) ще зависи от качеството на представените фалшификати.

За реализацията на режим идентификация са необходими единствено собствени подписи от всеки потребител. В този случай се създава и съхранява една обща невронна мрежа, която има толкова изходни неврони, колкото е броя на участниците. Изходът от идентификацията са толкова на брой стойности между нула и единица, колкото е и броят на всички потребители и като резултат се избира участника, съответстващ на максималната изходна стойност на мрежата. Когато се използва обща невронна мрежа, е необходимо тя да се обучи наново при добавяне на участник, което несъмнено изисква повече време.

3.1.6 Проектиране на невронни мрежи

Задачата за проектиране на невронна мрежа включва следните подзадачи : *събиране на данни, създаване на мрежата, конфигуриране на мрежата, инициализиране на теглата и отклоненията ѝ, обучение, оценка на мрежата и същинското ѝ приложение* [49].

1) *Събиране на данни* - Качеството на събраните данни и множеството от наличните входни данни (признаци), които постъпват на входа ѝ са от голямо значение. В практическите задачи не е известна взаимовръзката между входните данни и желания изход и поради това е желателно да се събере множество разнообразни признаци, като само част от тях действително да бъдат важни за класификацията;
2) *Създаване на мрежата* - Броят на елементите от входния слой зависи от размерността на входните данни, а броят на елементите от изходния слой - от броя на разпознаваните класове. Входът на невронната мрежа представлява множество входни стойности, организирани във входни вектори от признаци. За да може НМ да има добро качество, добра практика е [66] да се изберат некорелирани входни променливи и да се извърши тяхното нормиране. Без извършването на нормировка, съществува голяма вероятност обучаващата функция да попадне в локален минимум. При *z-score* нормировката (вж. 2.3.1) средната стойност на всеки признак, изчислено върху множеството от примерите за обучение, е равно на нула, а стандартното отклонение е равно на единица.
3) *Конфигуриране на мрежата* - При проектирането на невронната мрежа следва да се вземе решение относно нейната топология (брой на слоевете и брой на невроните във всеки слой, техния тип и свързаност) и нейните параметри. От една страна, изборът им е от изключително важно значение, а от друга страна не съществува алгоритъм за техния избор, понеже те са силно зависими от решавания проблем и наличните данни. В литературата могат да се открият някои насоки, които да ориентират при вземането на решение за топологията на мрежата и параметрите ѝ [66]. Определянето на параметрите на мрежата не е тривиално и не се подчинява на някакво множество от правила и затова стойностите им се намират експериментално.

   При проектирането на многослоен перцептрон с топология *I-H-O*, където с *I* и *O* са означени съответно броя на входните неврони и броя на невроните в изходния слой, с *H* – броя на невроните в скрития слой, а с *Ntrn* – броя на образците в

обучаващата извадка, чрез следните формули се намират стойностите на броя на обучаващите уравнения *Neq*, броя на неизвестните тегла *Nw* и броя на степените на свобода *Ndof* [63]:

$$\boldsymbol{Neq = Ntrn * O} \tag{3.3}$$

$$\boldsymbol{Nw = (I + 1) * H + (H + 1)} \tag{3.4}$$

$$\boldsymbol{Ndof = Neq - Nw} \tag{3.5}$$

Реалните данни обикновено са повлияни от шум, грешки при измерването, грешки при закръгляване и др. Следователно, ако броят на уравненията *Neq* се равнява на този на неизвестните тегла *Nw*, то НМ ще запомня определени примери и няма да обобщава добре при представянето на непознати примери. Оптималната стойност на съотношението $r = Neq/Nw$ е в интервала [2, 32], като се предпочита $r \geq 8$ [68]. За да е сходимо обучението по алгоритъма с обратно разпространение на грешката, е необходимо стойността на *H* да е достатъчно голяма, за да моделира вариациите във взаимовръзката вход-изход. За да се отрази закономерността в данните и да се намали влиянието на грешки при измерването и шума в данните, е нужно и стойността на *Ntrn* да е достатъчно голяма ($Neq >> Nw$). Ако условието $Neq >> Nw$ не е изпълнено, то мрежата е пренастроена, тъй като броят на неизвестни тегла надвишава този на ограничаващите уравнения. В този случай следва да се приложи една от следните техники: (1) регуляризация (чрез командата *trainbr* в MATLAB) или (2) по-ранно спиране на процеса на обучение с валидираща извадка [63].

Броят и размерността на скритите слоеве се определят експериментално в зависимост от конкретната задача. Наличието на повече от един скрит слой несъмнено изисква необходимостта от по-големи изчисления, но пък може да доведе до по-ефикасното решаване на сложни проблеми. Не съществува точно правило, по което да се пресметне броя на възлите в скрития слой. Оптималната архитектура на невронната мрежа (в т.ч. и броят на скритите неврони) се намира експериментално, чрез тестването на различни НМ и оценяването на способността за обобщаване на всяка една от тях [34]. Оптималният брой на невроните в скрития слой *H* зависи по комплексен начин от размерността на входа и изхода, размера на обучаващата извадка, архитектурата, обучаващия алгоритъм, вида на активационната функция на изходния слой. Ако броят им е твърде малък, ще има висока грешка върху обучаващата извадка и висока грешка на обобщаване поради недостатъчно настройване на модела (*underfitting*). Обратно, при

прекалено голям брой скрити неврони грешката върху обучаващата извадка е малка, но грешката на обобщаване ще продължава да е висока поради пренастройването на модела (*overfitting*). За определяне на размерността на скрития слой на невронните мрежи е следвана методологията от [63,64]. Ако стойността на $r \geq 1$, то горната граница *Hub* за кандидат стойностите на *H* се определя по следния начин:

$$\boldsymbol{Hub = (Ntrn - 1) * O/(I + O + 1)} \qquad (3.6)$$

В случай, че са необходими по-голям брой неврони в скрития слой, се използват техники за регуляризация и/ или по-ранно спиране на процеса на обучение, дори и ако $r < 1$.

4) *Инициализиране на теглата и отклоненията на мрежата*

   Тъй като успехът на проектираната НМ зависи от произволните стойности на началните тегла, е необходимо да се обучат няколко НМ с различни инициализации на теглата и да се избере тази, която демонстрира най-добро качество. За всяка стойност на *H* в [*Hmin*, *Hub*] се обучават *k* на брой невронни мрежи за различни инициализации на теглата, като *k* обикновено приема стойности между 5 и 10, а стойността на *Hmin* се определя от изследователя. Избира се тази топология на НМ, съответстваща на възможно най-малката стойност на *H*, за която грешката е минимална.

5) *Обучение* - След като топологията на мрежата е дефинирана, е необходимо да се избере вида на обучаващия алгоритъм и да се премине към нейното обучение, т.е. към определянето на подходящи стойности на теглата на връзките между отделните й елементи, така че да се оптимизира нейното качество. Конфигурирането и обучението на невронната мрежа изискват наличието на примерни данни. Различните модели (в т.ч. и невронните мрежи) се характеризират чрез отношението на броя на наблюденията и броя на свободните параметри (степените на свобода). За да може мрежата да отразява някаква закономерност в данните, броят на векторите, използувани при обучението *Neq*, трябва да е по-голям (желателно значително по-голям) от броя на степените на свобода *Nw* (теглата) в мрежата *Neq >> Nw*. Модел, при който броят на наблюденията надвишава този на свободните параметри се нарича преопределен (*overdetermined*). Ако *Neq = Nw*, то моделът се нарича точно определен модел, а иначе – неопределен (*underdetermined*) модел. След като моделът на невронна мрежа бъде създаден се провеждат експерименти с нея. В случай, че получените резултати от работата на мрежата не

са достатъчно точни, изследователят обикновено предприема някои от следните възможности:

- Увеличаване на броя на скритите слоеве;
- Увеличаване на броя на скритите неврони;
- Увеличаване на броя на данните в обучаващата извадка;
- Увеличаване на броя на входните стойности ;
- Експериментиране с друг обучаващ алгоритъм.

6) *Оценка на мрежата и приложение* – Точността на дадена НМ зависи в значителна степен от размера на входния вектор, данните от обучаващата извадка и от възможностите на обучаващия алгоритъм. За оценяване на мрежата може да служи оценката на грешката *PCTError*, отразяваща процентното съотношение на грешно класифицираните образци към общия брой образци. Тази оценка се пресмята за обучаващата, тестовата и валидиращата извадки.

3.2 Метод на *k*-най-близки съседи

Методът на *k*-най-близки съседи е непараметричен метод за класификация. Нека е дадена обучаваща извадка D, състояща се от образци $(\boldsymbol{x}, y) \in D$, където $\boldsymbol{x}$ е вектор от признаци за даден образец, а y е етикет на неговия клас. Нека е дефинирана мярка за разстояние или сходство за намиране на разстояния между образците и е зададена стойност на k – броя на разглежданите най-близки съседи. За класификацията на даден тестов образец $z = (\boldsymbol{x}', y')$ се пресмятат разстоянията или близостта между него и всеки един от известните образци $(\boldsymbol{x}, y) \in D$ и по този начин се намира списък D_z от най-близките му k на брой съседи, а съответните им класове служат за определяне на принадлежността на неизвестния образец към съответния му клас например по метода на мажоритарно гласуване (*majority voting*).

$$y' = \underset{v}{\operatorname{argmax}} \sum_{(x_i, y_i) \in D_z} I(v = y_i) \qquad (3.7)$$

където v е етикет на клас, y_i е етикет на клас на i-тия най-близък съсед, а функцията I връща стойност, равна на единица, ако стойността на аргумента й е *true* и стойност, равна на нула в противен случай.

Частен случай (k =1) е метода на най-близкия съсед, при който за етикет на клас за неизвестния образец се счита този на най-близкия му съсед.

Отношение към точността на класификацията по метода на *k*-най-близки съседи имат: (1) стойността на k, (2) избора на мярка за разстояние или сходство, както и (3)

метода за комбинирането на етикетите на класовете. В случай на класификация в два класа (дихотомия) се препоръчва нечетна стойност на k. Максимална стойност за k е $k_{max} = \sqrt{n}$ [13]. За определяне на стойността на k е удачно да се оцени точността на метода чрез LOOCV (*leave one out cross validation*) за множество от стойности на k и да се избере тази от тях, която води до най-висока точност [13]. Според естеството на данните би могло да се използват различни мерки за разстояние. Най-често използвано за количествени променливи е Евклидовото разстояние. За получаване на добра класификация се препоръчва премахване на признаците, силно отдалечени от групата данни и нормиране на останалите признаци [67].

Тъй като няма експлицитно обучение или моделиране (т.нар. *lazy* обучаващ алгоритъм), то построяването на класификатор *k*NN е лесно и бързо, но класифицирането на непознати образци (по-конкретно изчисляването на разстоянията) е сравнително скъпо за големи обучаващи множества.

Класификаторът по *k*-най-близки съседи (*k*NN) е интуитивен, ясен, разбираем и лесен за употреба, а прилагането му обикновено дава добри резултати. Той може да се актуализира бързо при добавяне на нови образци с техните етикети на класове.

3.3 Оценяване и сравняване на качеството на класификатори

След проектирането на даден класификатор (в частност невронни мрежи) е необходимо да се оцени неговото качество. Оценката на точността на класификатори, обучени с учител, е важна не само за предсказване на точността при бъдещото им използване, но също така и при избор на класификатор и комбинирането на класификатори. Традиционната практика е множеството от проектираните модели да се оценят и измежду тях да се избере този с най-малка грешка. Известно е, че при сравняването на качеството на различни модели, прилагани за конкретна задача, резултатът е валиден само и единствено за тази задача и съответната база от данни и това е следствие от *No Free Lunch Theorem* в областта на разпознаването на образи [67]. Различните модели се сравняват преди всичко по стойностите на грешките им, като в зависимост от същината на конкретната задача, някои фактори също оказват влияние в различна степен. Такива фактори са следните: време за обучение и тестване, необходима памет, необходимост от проверка и валидиране от експерти и др. Някои от техниките, използвани за оценката на точността (метод *Holdout* и *N*-кратна крос валидация) са посочени по-долу.

3.3.1 Оценка на класификатори

В случая на разпознаване в два класа, съответно *Positive* (положителни образци) и *Negative* (отрицателни образци) се намира стойност на праг, която да ги разграничава най-добре. Така, един образец попада в даден клас в зависимост от това дали резултата от разпознаването надхвърля стойността на прага (клас *Positive*) или не (клас *Negative*) [100]. Това разпределяне би могло да бъде вярно (*True*) или грешно (*False*). Така, за даден образец от клас *Positive*, ако изходът от класификацията е положителен, то съответният образец се явява истински положителен (*True Positive*, *TP*), а ако е отрицателен, то образецът е лъжливо отрицателен (*False Negative, FN*). Аналогично, даден образец от клас *Negative* би могъл да бъде истински отрицателен *(True Negative, TN*) или лъжливо положителен (*False Positive*, *FP*). Вероятността за грешка за *FP* се нарича грешка от I тип и се означава с α (в този случай вероятността за *TP* е 1- α). Вероятността за грешка за *FN* се нарича и грешка от II тип и се означава с β (в този случай вероятността за грешка *TN* е 1- β). Матрицата на класификация (*Confusion Matrix*, Таблица 3.2) представя възможните варианти за класификация. По главния й диагонал се намират истински предсказаните образци (*TP* и *TN*), а извън него са лъжливо предсказаните образци (*FP* и *FN*).

Таблица 3.2 Матрица на класификация

Действителен/Предсказан	**Positive**	**Negative**
Positive	*TP*	*FN*
Negative	*FP*	*TN*

TAR (*true accept rate, Sensitivity*) показва какъв е дяла на правилно класифицирани в клас *Positive* образци от общия брой действително принадлежащи към класа *Positive* образци, т.е. показва каква част от образците в клас *Positive* са разпознати от класификатора.

$$TAR = Sensitivity = TP/(TP+FN)*100; \quad (3.8)$$

FAR (false accept rate) от своя страна показва какъв е дяла на неправилно класифицираните в клас *Positive* образци от общия брой в действителност принадлежащи към класа *Negative* образци.

$$FAR = FP/(FP+TN)*100; \quad (3.9)$$

По аналогичен начин се дефинират и *TRR* (*true reject rate, Specifity*) и *FRR* (*false reject rate*):

$$TRR = Specifity = TN/(TN+FP)*100; \quad (3.10)$$

$$FRR = FN/(FN+TP)*100; \quad (3.11)$$

В литературата са познати следните мерки за оценка на качеството на класификатори на базата на матрицата на класификация:

$$AC = (TP+TN)/(TP+TN+FP+FN)*100;$$

$$Precision = TP/(TP+FP)*100; \quad (3.12)$$

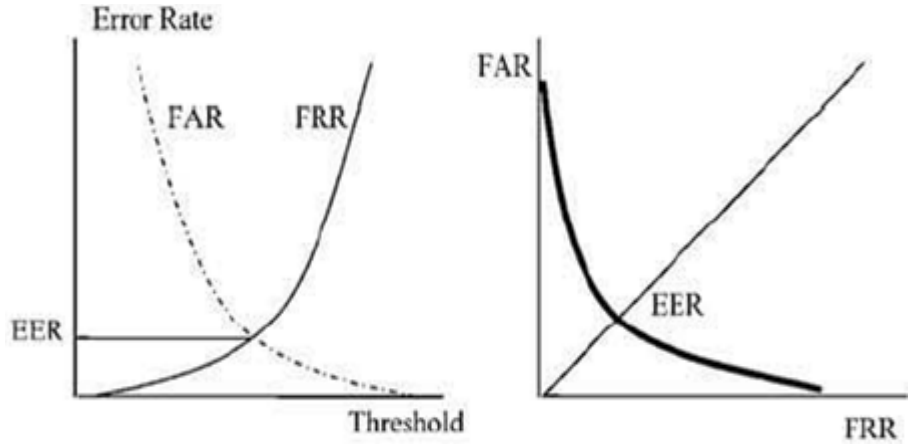
$$PCTError = (FP+FN)/(TP+TN+FP+FN)*100;$$

Точността на класификация (*classification accuracy*) *AC* се определя като съотношението между правилно класифицираните образци и общия брой образци. Прецизността на класификация (*Precision)* е мярка за точността на разпознаването и се дефинира като правилно класифицираните положителни образци (клас *Positive*). Грешката на класификация *PCTError* се намира като съотношението между погрешно класифицираните образци и общия брой образци.

За нагледно изобразяване на качеството на биометричните системи обикновено се изчертават т.нар. *Receiver Operating Characteristic* (ROC) криви, които представляват визуални средства за оценяване на точността на класификатор и за сравняване на различни класификационни модели. За получаването на траекторията на ROC кривата, за различни стойности на прага по вертикалната ос се изобразява стойността на *TAR*, а по хоризонталната ос - тази на *FAR*. Така, ROC кривата представлява крива на *TAR* (*Sensitivity*) срещу *FAR* (1-S*pecifity*). С преместването на прага от ляво на дясно, съответстващата му точка от ROC кривата също се отмества от дясно на ляво. При най-ниска стойност на прага, броят на лъжливо отрицателните образци (*FN)* и този на истински отрицателните образци (*TN)* обикновено е равен на нула и затова стойностите на *TAR* и на *FAR* ще са близки до единица. С нарастване на стойността на прага, броят лъжливо отрицателните образци (*FN)* и този на истински отрицателните образци (*TN)* намалява. Оптималното условие е *TAR* = 1 и *FAR* = 0, което отговаря на точка горе в

ляво на кривата. Алтернативен начин за представяне на качеството на системата е чрез стойността на оценката EER (*Equal error rate*). Тази оценка съответства на точката от ROC кривата, в която стойностите на *FRR* и на *FAR* са равни.

На следващата Фигура 3.2 са представени примерни визуални криви за оценяване на точността на класификатор: *FAR*, *FRR* и *EER*.



Фигура 3.2 FAR, FRR и EER

3.3.2 Метод *Holdout*

Методът *Holdout* се прилага при наличието на достатъчно примери за обучение. Множеството от образци се разделя в две извадки: за обучение и за тестване. Образците от обучаващата извадка се използват за определяне на параметрите на модела, а тези от тестовата извадка се използват за получаването на оценка на качеството на класификатора върху непознати и неизползвани за обучението му данни и за сравняването на различни мрежови архитектури. Тъй като не всичките данни се използват при обучението, то колкото е по-малка обучаващата извадка, толкова по-голяма ще е стойността на дисперсията на модела. От друга страна, ако размера на обучаващата извадка е много голям, то оценената точност върху по-малко тестово множество ще е непредставителна.

Прилагането на метода *Holdout* в случая на невронни мрежи се състои в следното. Образците се разделят в [49]: обучаваща, валидираща и тестова извадки. Данните от обучаващата извадка се представят по време на обучението и се използват за настройване на теглата на невронните мрежи. Тези от валидиращата извадка се използват за измерване на качеството на всяка от проектираните невронни мрежи върху непознати данни и при взимането на решение за прекратяване на нейното обучение, когато качеството престане да се подобрява. Данните от тестовата извадка се използват

за симулиране на невронната мрежа и измерване на нейното качество след приключване на обучението й. Така, кандидат невронните мрежи се нареждат във възходящ ред по качеството им, изчислено върху валидиращата извадка, а крайната грешка се смята върху данните от тестовата извадка. По подразбиране трите извадки се разделят в съотношение 0,7: 0, 15: 0,15. Най-общо, при прилагането на метода *Holdout* за НМ, се извършват следните стъпки (Фигура 3.3) [67]:

(1) Образците се разделят в обучаваща, валидираща и тестова извадки;

(2) Избира се архитектура на НМ и обучаващи параметри;

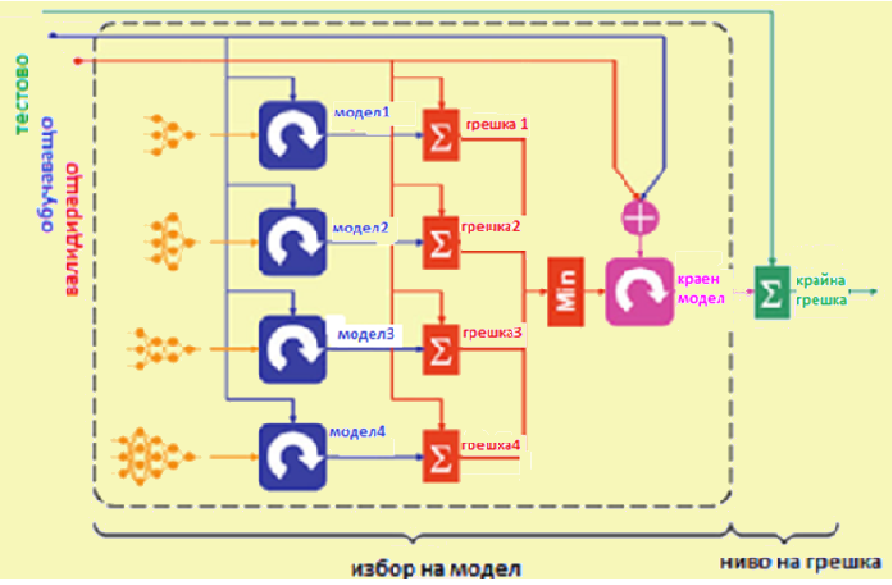
(3) Моделът на НМ се обучава с данните от обучаващата извадка;

(4) Моделът на НМ се оценява чрез данните от валидиращата извадка;

(5) Стъпките 1-4 се повтарят за различни архитектури и обучаващи параметри;

(6) Избира се най-добрия модел, който се обучава с данните от обучаващата и валидиращата извадки;

(7) Избраният модел се оценява чрез тестовата извадка;

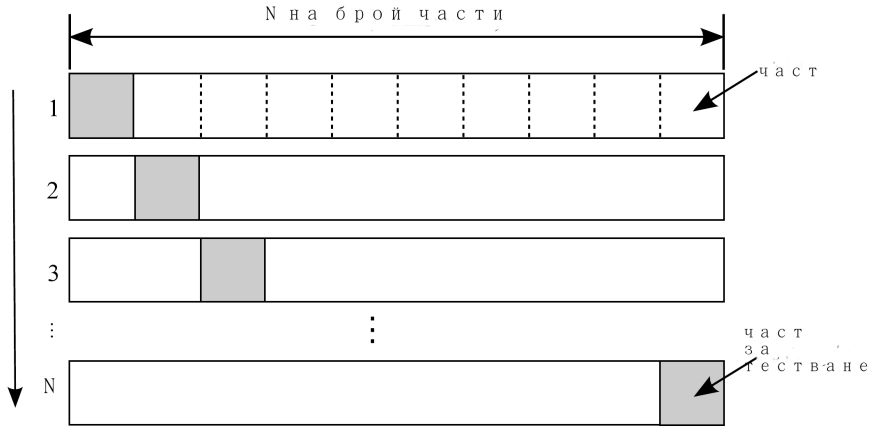


Фигура 3.3 Метод Holdout

3.3.3 Метод на *N*-кратна крос валидация (*N*-fold cross validation)

Методът на *N*-кратна крос валидация е приложим в случаите, когато на разположение са малък брой примери за обучение. Същността на метода се състои в

следното. Данните се разделят произволно на *N* части (*folds*) с приблизително равен брой образци във всяка част. Последователно се съставят всичките *N* на брой възможни извадки, всяка от които съдържа обучаваща извадка, състояща се от (*N-1*) части и тестова извадка, съдържаща останалата една част. По този начин всяка част се използва за тестване точно един път. С усредняване на стойностите на грешките, получени за всяко разделяне на данните, се намира оценка на стойността на крайната грешка на класификатора. Характерно за този метод е това, че за обучението на класификатор се използват всичките налични примери за обучение и по този начин не се губят никакви данни. Методът на *N*-кратна крос валидация е изобразен схематично на следната Фигура 3.4.



Фигура 3.4 Метод на N-кратна крос-валидация

Ако стойността на *N* е равна на размера на обучаващата извадка, то методът се нарича *Leave One Out Cross Validation (LOOCV)*. В този случай, тестовата извадка се състои от единствен образец, а за обучението се използват възможно най-много данни. Този метод често е изчислително скъп [67].

Изборът на стойност на *N* е от съществено значение. Ако стойността на *N* е висока, то стандартното отклонение на оценяваната точност би имало ниска стойност (моделът би бил точен), но това е за сметка на дисперсията на оценяваната точност (която би била висока) и на времето за изчисление. Обратно, ако стойността на *N* е ниска, то стандартното отклонение на оценяваната точност би имало висока стойност, а дисперсията й – ниска. В повечето случаи се избира $N = 10$. За големи бази от данни, дори и 3-кратната крос валидация би била достатъчна, докато за такива с ограничен обем данни, е удачно прилагането на *LOOCV,* за да се осигури обучение по възможно

най-голям брой обучаващи примери [67].

С прилагане на метода на *N*-кратна крос валидация може да се провери способността за обобщаване на невронната мрежа [49, 50] в случаите, когато на разположение са малък брой примери за обучение. Така, стъпки (3) и (4) от метода *Holdout*, описан в Раздел 3.3.2 се повтарят за всяко разделяне на данните.

Невронната мрежа се обучава за всяко от тези *N* на брой множества с *K* на брой инициализации на началните тегла на връзките. По този начин за всяко разделяне $i = 1, \dots, N$ се намират *K* на брой оценки на грешките $e_{i,j}$ $j = 1, \dots, K$, а стойността на крайната грешка за всяко разделяне E_i се изчислява като средната стойност от тези *K* на брой оценки:

$$E_i = \frac{1}{K}\sum_{j=1}^{K} e_{i,j,} \qquad i = 1, \dots, N; j = 1, \dots, K \tag{3.13}$$

Оценките E_i на съответните класификатори са неизместени, тъй като данните от тестовата извадка не се съдържат в обучаващата извадка. Оценката на конкретния модел $\hat{E}$ се намира като средната стойност от грешките, получени за всяко разбиване на данните:

$$\hat{E} = \frac{1}{N}\sum_{i=1}^{N} E_i \tag{3.14}$$

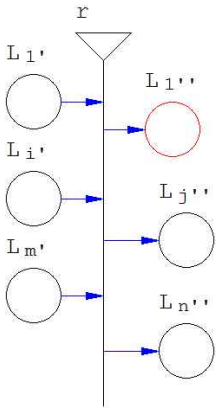
Оценката $\hat{E}$ разчита на допускането, че точността на класификацията би се променила плавно с промяна в размера на обучаващата извадка. Следователно, ако точността на модела е задоволителна, изследователят би могъл или да обучи единствен класификатор върху цялото множество от данни, или да комбинира класификаторите, построени по време на обучението [91].

Методът на *N*- кратна крос валидация е удачен избор при търсенето на подходяща стойност на *k* в класификатора *k*NN, а също и при търсенето на оптимална топология на НМ [13]. След намирането на оптимални стойности на параметрите на класификатор, той се обучава върху цялото множество от данни и бива използван на практика.

3.4 Обобщен мрежов модел за он-лайн верификация на подписи

Обобщените мрежи (ОМ) представляват разширение на мрежите на Петри и са инструмент за моделиране на паралелни и конкурентни процеси. Концепцията на ОМ е описана в [105]. Преходите в ОМ представят дискретни събития, а позициите – условия на конкретното събитие. Ядрата на ОМ се придвижват през преходите от една позиция в

друга, а информацията им се съхранява в множество от характеристики. Редуцирана обобщена мрежа е ОМ, от която са премахнати някои компоненти. Тя съдържа преходи (общият им вид е показан на Фигура 3.5) от вида $Z = \langle L', L'', r, \square \rangle$, където:



Фигура 3.5 Общ вид на преход в ОМ

1) L' и L'' са крайни, непразни множества от съответно входни и изходни позиции. $L' = \{L'_1, \dots, L'_i, \dots, L'_m\}$, $L'' = \{L''_1, \dots, L''_j, \dots, L''_n\}$.
2) r представя условията на прехода във вид на индексирана матрица:

r =		L''_1 ... L''_j ... L''_n
	L'_1	
	L'_i	r_{ij} ($1 \le i \le m; 1 \le j \le n$ r_{ij} −предикат
	L'_m	

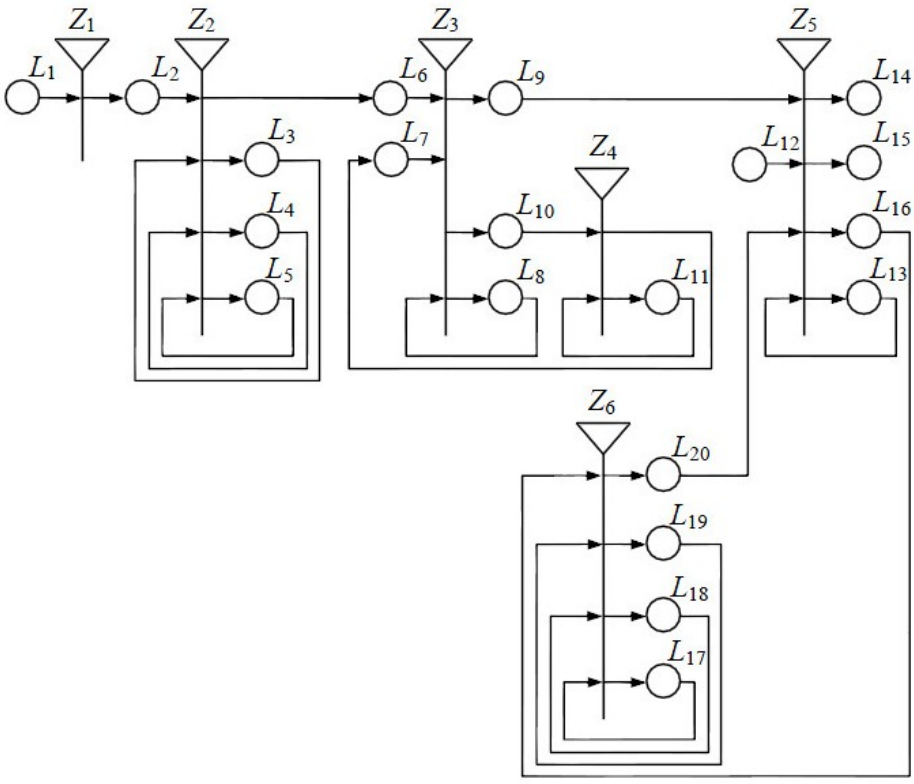
   В случай, че стойността на предиката r_{ij} е *True* ядрото от позиция i може да премине на позиция j, като в противен случай това е невъзможно. Най-общо казано матрицата от предикати описва условията за движение на ядрата в различните посоки.
3) $\square$ представлява булев израз, чиято стойност (*True* или *False*) определя дали прехода може да стане активен.

Наредената четворка $E = \langle A, K, X, \Phi \rangle$ се нарича редуцирана обобщена мрежа, ако:

i. A е множество от преходи;
ii. K е множество от ядра;
iii. X е множеството от начални характеристики, които ядрата получават с постъпването си в мрежата;

iv. Ф е характеристична функция, която задава характеристики на ядрата при преминаването им през даден преход.

С апарата на обобщените мрежи (ОМ) е направен опит за моделиране на процеса на верификация по подпис с цел ясно и нагледно представяне на предложения в дисертацията метод, който включва следните стъпки за верификация на заявена самоличност по подпис: полагане на подпис, предварителна обработка на подпис, извличане на признаци, избор на признаци, обучение и верификация. Представената редуцирана обобщена мрежа е без времеви компоненти, без зададени приоритети на преходите, позициите и ядрата и представлява модификация на предложените в [103, 104] модели на ОМ за различни задачи от разпознаването на образи. Обобщено мрежовият модел е представен на Фигура 3.6. Той е изграден от 6 прехода (всеки един от които представя стъпка от процеса), 20 позиции и 3 вида ядра α, β и γ.



Фигура 3.6 Обобщен мрежов модел на процеса за он-лайн верификация по подпис

Първоначално в позиция L_1 постъпва α-ядро с начална характеристика: „*Положен он-лайн подпис за верификация”*. По време на цялото функциониране на модела в позиция L_7 присъства β - ядро с начална и текуща характеристика “*Подмножество от признаци, получени от базата данни за подписи*”. Всички постъпващи в позиция L_7 ядра се сливат с оригиналното β -ядро. По време на цялото функциониране на модела на

позиция L_{12} има ядро γ с начална характеристика "*Съхранен модел на невронна мрежа за конкретния потребител*". Всички постъпващи в позиция L_{12} ядра се сливат с оригиналното γ -ядро. За краткост ще се използва означението α, β, γ -ядра като няма да се използват индекси, указващи текущия номер на съответното ядро.

Входна позиция на мрежата е L_1 като през нея в модела постъпва α -ядро с начална характеристика: „*положен он-лайн подпис за верификация"*.

Преходът $Z1$ представя в модела стъпката полагането на он-лайн подпис.

$$Z_1 = \langle \{L_1\}, \{L_2\}, r_1, \square_1 \rangle$$

$r_1 =$

	L_2
L_1	$True$

Условието за настъпване на прехода е: $\square_1 = \vee (L_1)$.

Ядрото α получава характеристика "*получени характеристики/ данни за всяка точка от подписа*" в позиция L_2.

Преходът Z_2 описва предварителната обработка на подписа, която включва последователното прилагане на трансформация на координатите, ротация и транслация.

$$Z_2 = \langle \{L_2, L_3, L_4, L_5\}, \{L_3, L_4, L_5, L_6\}, r_2, \square_2 \rangle$$

Условието за настъпване на прехода е: $\square_2 = \vee (L_2)$.

$r_2 =$

	L_3	L_4	L_5	L_6
L_2	$True$	$False$	$False$	$False$
L_3	$W_{3,3}$	$W_{3,4}$	$False$	$False$
L_4	$False$	$W_{4,4}$	$W_{4,5}$	$False$
L_5	$False$	$False$	$W_{5,5}$	$W_{5,6}$

Предикатите, асоциирани с прехода са следните:

$W_{3,4}$= "Извършена е трансформация на координатите на подписа".

$W_{3,3}$= ┐ $W_{3,4}$.

$W_{4,5}$= " Извършена е ротация на подписа".

$W_{4,4}$= ┐ $W_{4,5}$.

$W_{5,6}$=" Извършена е транслация на подписа".

$W_{5,5}$=┐$W_{5,6}$.

Ядрото α получава характеристика "*данните за подписа са предварително обработени*" в позиция L_6.

Преход Z_3 описва стъпката извличане на признаци за подпис. Той се задава чрез:

$$Z_3 = \langle \{L_6, L_7, L_8\}, \{L_8, L_9, L_{10}\}, r_3, \square_3 \rangle$$

$r_3 =$

	L_8	L_9	L_{10}
L_6	$W_{6,8}$	$False$	$W_{6,10}$
L_7	$W_{7,8}$	$False$	$False$
L_8	$W_{8,8}$	$W_{8,9}$	$False$

От базата данни се получава информация за подмножеството от избрани признаци за конкретния потребител. Затова и ново ядро β постъпва в позиция L_7 с начална характеристика "*Подмножество от признаци, получени от базата данни за подписи*". Въз основа на характеристиките на подписа се изчисляват конкретните признаци последователно в позиция L_8.

Условието за настъпване на прехода е: $\square_3 = \wedge (L_7, \vee (L_6, L_8))$.

Предикатите, асоциирани с прехода са следните:

$W_{6,8}$=$W_{7,8}$="Съществува избрано подмножество от признаци за потребителя в БД".

$W_{6,10}$= ┐$W_{6,8}$.

$W_{8,9}$= " Стойностите на всички признаци от избраното подмножество и общото подмножество са извлечени".

$W_{8,8}$=┐$W_{8,9}$.

Ядрото α от позиция L_6 се слива с ядрото β от L_7 в ново α – ядро в позиция L_8. Ядрото α в позиция L_9 получава характеристика:"*Извлечени са стойностите на всички необходими признаци*", а това в позиция L_{10}: "*Не е извършван избор на признаци за потребителя*".

Преход Z_4 се отнася за избор на общи признаци и на признаци по *Вариант 1* и записът им в БД:

$$Z_4 = \langle \{L_{10}\}, \{L_{11}\}, r_4, \vee (L_{10}) \rangle$$

$r_4 =$

	L_7	L_{11}
L_{10}	$True$	$False$
L_{11}	$False$	$True$

Ядрото α в позиция L_{10} получава характеристика "*Извършен е избор на признаци от БД в позиция* L_{11}" и се слива с оригиналното β в позиция L_7.

Преход Z_5 описва стъпката верификация с невронна мрежа. Извършва се тестване на съхранената за потребителя невронна мрежа (ако такава вече съществува) със стойностите на съответните признаци. Заявената самоличност с подписа се приема, ако изходът от верификацията надвишава съответния праг и се отхвърля в противен случай.

$$Z_5 = \langle \{L_9, L_{12}, L_{13}, L_{20}\}, \{L_{13}, L_{14}, L_{15}, L_{16}\}, r_5, \square_5 \rangle$$

$r_5 =$

	L_{13}	L_{14}	L_{15}	L_{16}
L_9	$W_{9,13}$	$False$	$False$	$W_{9,16}$
L_{12}	$W_{12,13}$	$False$	$False$	$False$
L_{13}	$False$	$W_{13,14}$	$W_{13,15}$	$False$
L_{20}	$W_{20,13}$	$False$	$False$	$False$

Ново ядро γ постъпва в позиция L_{12} с начална характеристика "*Съхранен модел на невронна мрежа за конкретния потребител*".

Условието за настъпване на прехода е: $\wedge (L_{12}, \vee (L_9, L_{13}))$.

Предикатите, асоциирани с прехода са следните:

$W_{9,13}$=$W_{12,13}$=$W_{20,13}$ ="Съществува създаден модел на НМ за потребителя".

$W_{9,16}$= $\neg W_{9,13}$.

$W_{13,14}$= "Тестване на модела: Заявената самоличност е потвърдена".

$W_{13,15}$= "Тестване на модела: Заявената самоличност е отхвърлена".

Ядрото в позиция L_{14} получава характеристика "*Самоличността е потвърдена*". Ядрото в позиция L_{15} получава характеристика "*Самоличността е отхвърлена*". Ядрото на позиция L_{16} запазва всичките си характеристики до момента.

Преход Z_6 описва тестването на различни модели на НМ, избор на модел и параметри на невронна мрежа и нейното обучение. Избраният чрез крос валидация модел се запаметяма наред с входното множество признаци и стойността на прага.

$$Z_6 = \langle \{L_{16}, L_{17}, L_{18}, L_{19}\}, \{L_{17}, L_{18}, L_{19}, L_{20}\}, r_6, \square_6 \rangle$$

$r_6 =$

	L_{17}	L_{18}	L_{19}	L_{20}
L_{16}	$True$	$False$	$False$	$False$
L_{17}	$W_{17,17}$	$W_{17,18}$	$False$	$False$
L_{18}	$W_{18,17}$	$True$	$W_{18,19}$	$False$
L_{19}	$False$	$False$	$False$	$True$

Условието за настъпване на прехода е: $\wedge (L_{16}, \vee (L_{17}, L_{18}))$.

Предикатите, асоциирани с прехода са следните:

$W_{17,17}$= "Необходимо е да се създаде модел на НМ с номер *s*+1, където *s* е броя на циклите, които ядрото е извършило до момента на тази позиция, *s*<N, N – общ брой на моделите".

$W_{17,18}$= "Създаденият на стъпка *s+1* модел не е оценен чрез крос валидация, където *s* е броя на циклите, които ядрото е извършило до момента на тази позиция".

$W_{18,17}$= "Ако $s < N$".

$W_{18,19}$= $\neg W_{18,17}$.

Ядрата, постъпващи в позиция L_{18} получават характеристика "*Оценка на текущия модел, получена чрез крос валидация*". Ядрото, постъпващо в позиция L_{19} получава характеристика "*Избран е най-добрият модел според получените оценки от крос валидация*". Ядрото, постъпващо в позиция L_{20} получава характеристика "*Обучен е най-добрият модел на НМ и е запаметен*" и се слива с оригиналното γ ядро в позиция L_{12}.

3.5 Резултати в Глава 3

1. Представена е кратка класификация на различни видове невронни мрежи според архитектурата и алгоритмите за обучение. Коментирани са предимствата и недостатъците им.
2. Анализирани са въпросите за изграждане и обучение на невронни мрежи.
3. Описани са видовете грешки от класификация, методите за оценка на точността и построяването на комбиниран класификатор.
4. Изграден е модел на процеса на он-лайн верификация с апарата на обобщените мрежи.

Глава 4. Експерименти и резултати

В настоящата глава са представени получените резултати от провеждането на експериментите за създаване и тестване на комбинираната система за разпознаване по подпис. Тази система може да служи за верификация въз основа на признаците, описани в Раздел 2.2. За решаване на поставената задача са проектирани класификатори невронни мрежи от тип многослоен перцептрон и класификатори по *k*-най-близки съседи, които са тествани върху образците на подписи от две бази данни**.** Комбинираният подход за разпознаване по подпис се прилага като поредица от следните стъпки: (1) предварителен избор на признаци и (2) верификация.

С провеждане на експериментите се цели да се проверят следните предположения:

1) Задачата за избор на подмножество от признаци може да се реши чрез прилагането на метод за избор на регресионни променливи, базиран на критерия *Cp* на *Mallows*, с който може да се намери специфично най-информативно подмножество от признаци за всеки от участниците. Тези подмножества от признаци ще се търсят по два начина. При първия се използват собствените подписи и неумели фалшификати, а при втория собствените подписи и умели фалшификати. Получените подмножества от признаци ще се сравнят;
2) Резултатите от верификация по редуцираното множество от признаци ще са по-добри от тези по общото множество от признаци;
3) Резултатите от верификация по редуцираното множество от признаци ще са по-добри с използване на умели и неумели фалшификати за обучение на класификатора, отколкото само с използване на неумели фалшификати.

4.1 Избор на признаци за разпознаване

За всеки от потребителите се извличат стойностите на признаците, изчислени по характеристиките на всичките му осем или десет собствени и десет подправени подписа, записани в таблицата *StrokeInSignatureOriginalData* (Фигура 5.2). В настоящата дисертация е експериментирано с умели и неумели фалшификати. Неумелите фалшификати са избрани произволно от собствените подписи на останалите участници. Признаците са глобални и са описани в раздел 2.2. Тъй като всеки от тях получава стойности в различни интервали при различна скала на измерване, се налага необходимостта от нормиране на признаците в интервала [0,1]. Извършва се нормировка *min-max* (раздел 2.3.1), която запазва взаимовръзките между оригиналните стойности на данните. След извличането на признаци, се пристъпва към намирането на (1) ***общо*** за всички потребители подмножество от признаци с прилагане на метода на

корелационните плеяди (точка 2.3.2); (2) ***индивидуално*** подмножество от признаци на базата на собствени подписи и ***неумели*** фалшификати (наричано по-долу Вариант "1"); (3) ***индивидуално*** подмножество от признаци на базата на собствени подписи и ***умели*** фалшификати (наричано по-долу Вариант "2");. Така намерените подмножества от признаци служат за построяването на класификационните модели.

След намирането на общо за всички потребители подмножество от признаци, се прилага метода за избор на регресионни променливи, описан в Раздел 2.3 за намирането на индивидуалните подмножества от признаци (по Вариант "1" и Вариант "2"). Така се откриват най-добрите подмножества от признаци с различен размер за всеки участник въз основа на неговите собствени и подправени подписи. Сред тях се избира това подмножество от признаци, за което стойността на *Cp* е най-близка до тази на *p*, където *p* е броят на регресионните коефициенти. Получените по този начин подмножества от признаци са с различен размер за различните потребители. Нека означим броя на признаците с *k*, а този на подписите на даден потребител с *n*. За всеки потребител се създава текстов файл с двадесет реда, равен на броя подписи, определени за обучение. Всеки ред съдържа стойности на признаците за съответния подпис, разделени с точка и запетая, следвани от 1 (ако конкретният подпис е собствен, тоест принадлежи на първия клас) и -1 (ако подписът е подправен, тоест принадлежи на втория клас). Необходимо е да се подберат тези признаци (независими променливи), които възможно най-добре апроксимират наблюдаваните значения на зависимата променлива *y* ($y = 1$ за точките от първия клас, собствен подпис и $y = -1$ за тези от втория клас, фалшифициран подпис). Този файл се обработва и за всеки потребител / файл се откриват най-добрите подмножества с размер *p* ($p = k - r$) за стойности на *r* ($3 < r < 13$), а сред тях се намира подмножеството със стойност на *Cp* най-близка до *p,* но по-малка от нея. Това подмножество от признаци се приема за "най-добро".

4.2 Построяване на модели за класификация

Експериментите са извършени с класификатори невронни мрежи и класификатор по *k*-най-близки съседи. Построени са класификатори с вариращи параметри (брой признаци, вид фалшифицирани подписи за обучение, стойност на *H* при НМ и стойност на *k* при *k*NN). Поради малкия обем данни за обучение, точността на различните модели за класификация е оценена с 10-кратна крос валидация (НМ) и LOOCV (*k*NN).

4.2.1 Класификатор невронни мрежи

За всеки потребител се създава отделна НМ. За създаването на невронните мрежи е следвана методологията, описана в Раздел 3.1.3. Входът *p* на всяка от тестваните невронни мрежи представлява матрица [*FeatureSize Ntrn*] с брой редове, равен на броя на признаците *FeatureSize* и брой стълбове, равен на броя на образци за обучение *Ntrn*. Изходът зависи от режима на разпознаване. В случай на верификация той е матрица с размерност [*1 Ntrn*], със стойност в [0,1] във всеки от редовете, равна на 1 за собствен подпис и 0 – за подправен.

В процеса на обучение участват и собствени и подправени подписи. Експериментирано е с използването само на неумели фалшификати (наричано по-долу Случай ”1”) и едновременно на умели и неумели фалшификати (наричано по-долу Случай ”2”). При определянето на броя подписи за обучение се ръководим от следното правило. Ако с N1 означим броя на положителните примери (клас “1”), а с N0 – тези на отрицателните (клас “0”), то N1/N0 би трябвало да принадлежи на интервала [0.5, 2], [63]. В противен случай (ако броят на отрицателните примери значително надвишава този на положителните) грешната класификация на собствени подписи оказва незначително влияние върху формирането на общата грешка на класификатора. Тъй като броят на събраните подписи от всеки потребител е малък, то не е практично част от тях да се използват за валидация и затова се използва алтернативен обучаващ алгоритъм Бейсова регуляризация чрез алгоритъма *trainbr* на MATLAB. В търсене на оптимална архитектура на НМ, за всеки потребител се експериментира с различни топологии на един и същ вид невронна мрежа според променливия размер на броя на невроните в скрития слой. Топологията на НМ се избира чрез крос валидация.

Така за всеки потребител се намират параметри на модел на НМ (брой скрити неврони, вид фалшиви подписи за обучение, брой признаци на входа и др.). Изходът от верификацията е стойност от интервала [0,1]. За всеки участник се определя т.нар. праг на приемане (*score threshold*). Ако резултатът от верификацията надвишава този праг, съответният подпис се счита за автентичен, а в противен случай – за подправен. Окончателната невронна мрежа се запаметява заедно с архитектурата и стойностите на теглата и може да се използва по-нататък за верификация.

4.2.2 Класификатор по k-най-близки съседи

За създаването на класификатор по *k*-най-близки съседи са следвани препоръките, изложени в Раздел 3.2. Първоначално се прилага нормиране на признаците, така че те да имат стандартно отклонение, равно на единица и средна стойност, равна на нула.

Използвано е Евклидово разстояние за пресмятане на разстоянията между образците. Тъй като $k_{max} = \sqrt{Ntrn}$, $Ntrn$ – брой примери за обучение и стойността на k е препоръчително да е нечетно число, то са тествани са две стойности на k: 1 и 3.

4.3 Получени резултати върху собствена база данни за подписи

Намиране на общо подмножество от признаци

Началният брой признаци е 26. За всеки потребител, за всяка двойка признаци, е пресметната стойността на взаимната им корелация (вж. Раздел 2.3). В Таблица 4.1 са отразени броят на срещанията (в случай че се среща при повече от 25% от потребителите) на признаците, за които се наблюдава висока корелация за всичките 8 потребители.

Таблица 4.1 Брой срещания на двойки признаци с висока взаимна корелация

Признак 1	Признак 2	Брой срещания
A5	A4	8
A14	A1	7
A15	A1	7
A15	A14	7
A7	A3	6
A10	A9	6
A19	A1	6
A3	A2	5

Признак 1	Признак 2	Брой срещания
A19	A9	5
A19	A14	5
A19	A15	5
A9	A2	4
A10	A1	4
A10	A2	4
A18	A14	4
A19	A10	4

Въз основа на данните от Таблица 4.1, с прилагане на метода на корелационните плеяди [99], премахваме от по-нататъшно разглеждане признаците A1, A3, A4, A7, A9, A10, A14, A18, A19. Признаците, които остават са 17 на брой и са следните: A2, A5, A6, A8, A11, A12, A13, A15, A16, A17, A20, A21, A22, A23, A24, A25, A26. Те формират общото подмножество от признаци.

Намиране на индивидуално подмножество от признаци за всеки потребител

Резултатите от избора на най-информативните признаци по метода на *Hocking* и *Leslie* (Раздел 2.3) за всеки от осемте потребители са обобщени в Таблица 4.2. Със знак "+" са означени номерата на признаците, които участват в така намерените индивидуални подмножества. Вариант "1" се отнася до случая, при който подмножеството от индивидуални признаци се определят от 10 собствени подписа и 10 неумели фалшификати, а вариант "2" – от 10 собствени подписа и 10 умели

фалшификати. От таблицата е видно, че в случая на вариант “2” броя на признаците за всички потребители без един се редуцира до 13, като само за потребител #5 те са 5 признака, докато при вариант “1” броя на признаците за половината участници се редуцира до брой, по-малък от 13.

Таблица 4.2 Редуцирани признаци за всеки потребител

Участник	**#1**		**#2**		**#3**		**#4**		**#5**		**#6**		**#7**		**#8**	
Вар.	“1	“2	“1”	“2	“1	“2”	“1	“2	“1	“2	“1”	“2”	“1”	“2”	“1”	“2”
A2	+	+	+			+		+	+			+	+	+	+	
A5		+	+	+	+	+	+	+	+	+	+		+	+		+
A6	+		+	+		+		+	+	+	+	+			+	+
A8		+	+	+		+	+	+	+	+		+	+		+	
A11	+	+				+					+			+	+	+
A12	+			+		+		+	+			+	+	+	+	+
A13	+	+		+	+	+		+			+	+	+	+		+
A15	+	+	+	+		+		+			+	+	+	+		
A16			+	+	+	+		+	+		+	+	+	+	+	
A17		+	+	+	+	+	+	+	+		+	+		+		+
A20	+	+	+	+	+		+		+		+			+		+
A21	+	+	+	+	+	+	+	+	+		+	+	+	+	+	+
A22	+	+						+	+	+	+	+	+	+	+	+
A23	+	+	+		+		+				+		+		+	+
A24	+	+	+	+	+	+	+	+			+	+	+		+	+
A25	+		+	+					+			+	+	+		+
A26	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
Ср	10,	9,0	9,09	9,5	8,1	9,05	7,1	9,2	10,	4,8	9,08	9,33	9,12	9,24	10,	9,77
р	13	13	13	13	9	13	8	13	12	5	13	13	13	13	11	13

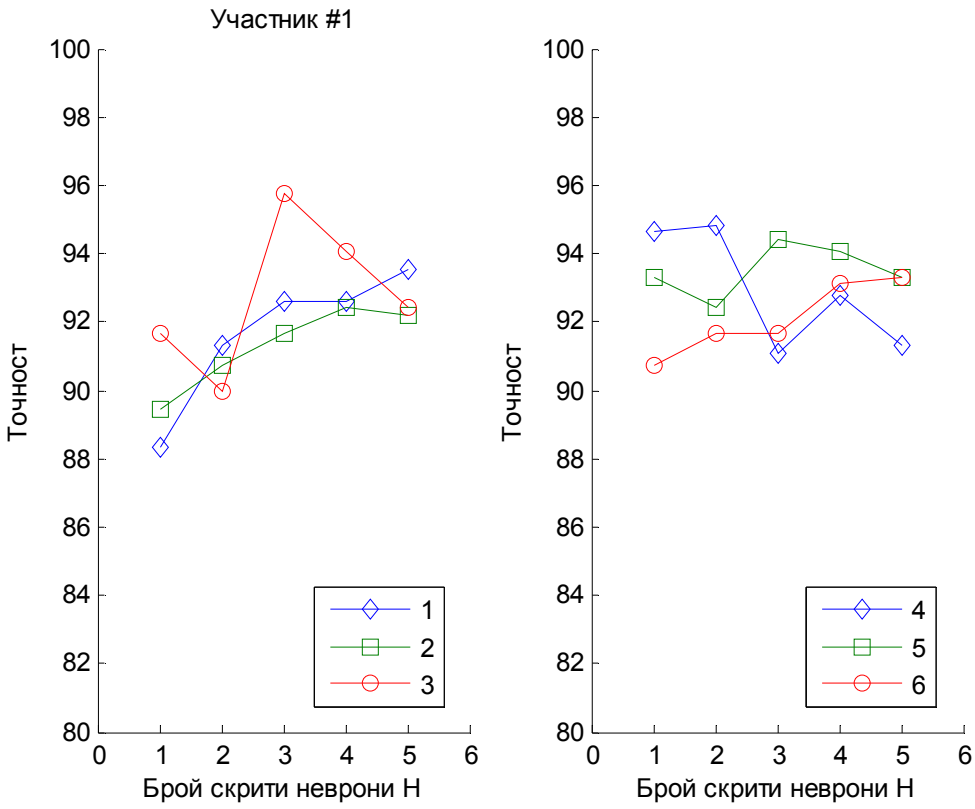
Таблица 4.3 Параметри на моделите

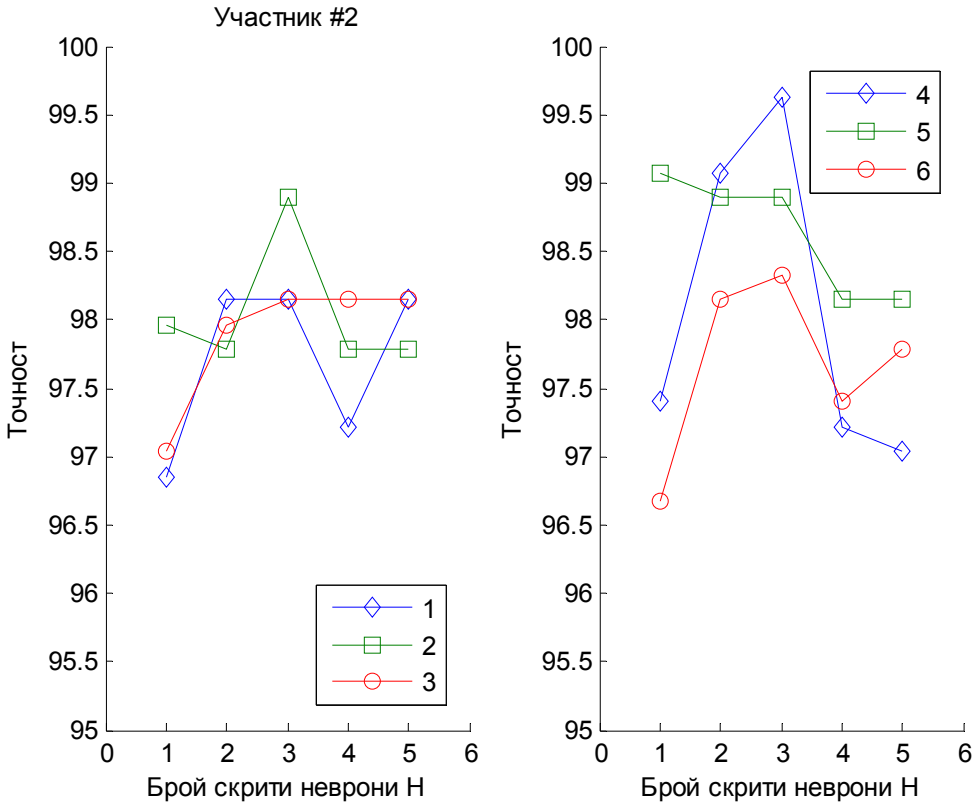
# на модел	**Брой признаци**	**Брой собствени подписи**	**Брой подправени подписи**		**Брой скрити неврони *H***	**Брой съседи *k***
1	Общи признаци	10	15 неумели	Случай “1”	от 1 до 5	1 или 3
2	Вариант “1”					
3	Вариант “2					
4	Общи признаци		9 неумели и 6 умели	Случай “2”		
5	Вариант “1”					
6	Вариант “2”					

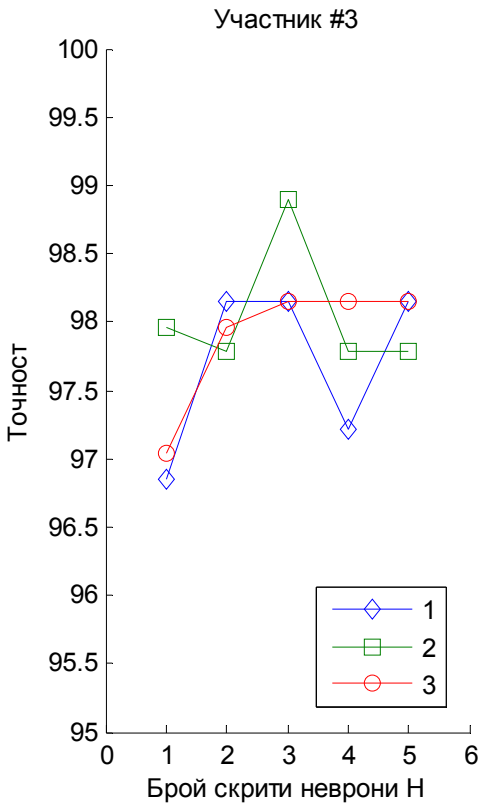
Таблица 4.3 съдържа обобщена информация за всеки от построените модели (с номера от 1 до 6).

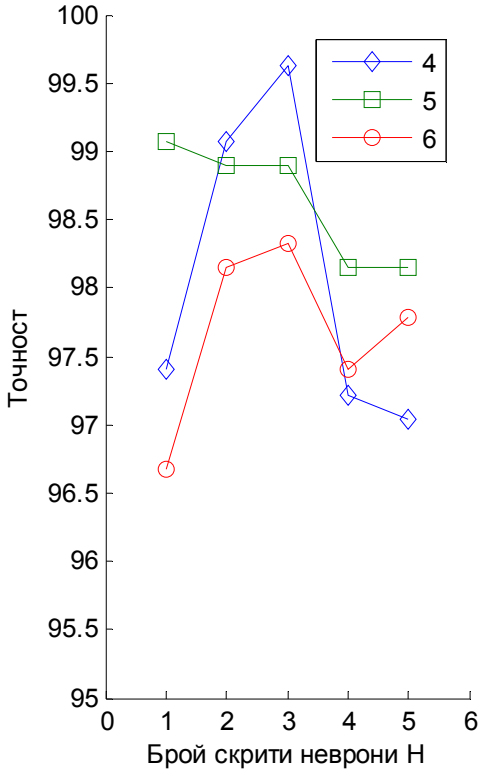
Проектираните невронни мрежи за всеки участник са с H=1-5 брой неврони в скрития слой. Общият им брой е 30. Те се оценяват и сравняват чрез метода на крос валидация (Раздел 3.3.3). Броят примери за обучение Ntrn е 25, този на обучаващите уравнения Neq е 25, а горната граница за брой скрити неврони Hmax = 1. За да може да се експериментира с различни стойности на H като не се налага ограничение за стойността на H и за да се избегне пренастройване, окончателните модели на невронни мрежи за всеки потребител се обучават с алгоритъма Бейсова регуляризация trainbr.

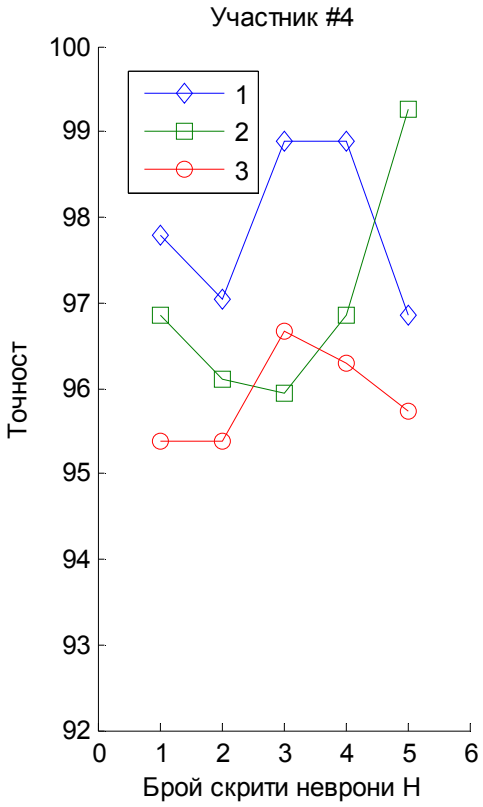
Фигурите по-долу изобразяват получените стойности на приближените оценки на точността на НМ за различни стойности на брой на скритите неврони H (H=1,…,5) за всеки един от участниците. Построените модели се различават по използваните отрицателни примери за обучение и по броя на признаците. На фигурата в ляво са изобразени данните за модели, построени върху обучаваща извадка, състояща се от собствени подписи и неумели фалшиви подписи (Случай “1”), докато тази в дясно се отнася до модели, обучени с собствени подписи, умели и неумели фалшифиви подписи (Случай “2”). Със син цвят са означени моделите, построени върху 17 броя общи признаци, със зелен цвят са означени тези, построени върху признаците от Вариант “2”, а с червен – тези от Вариант “1”.

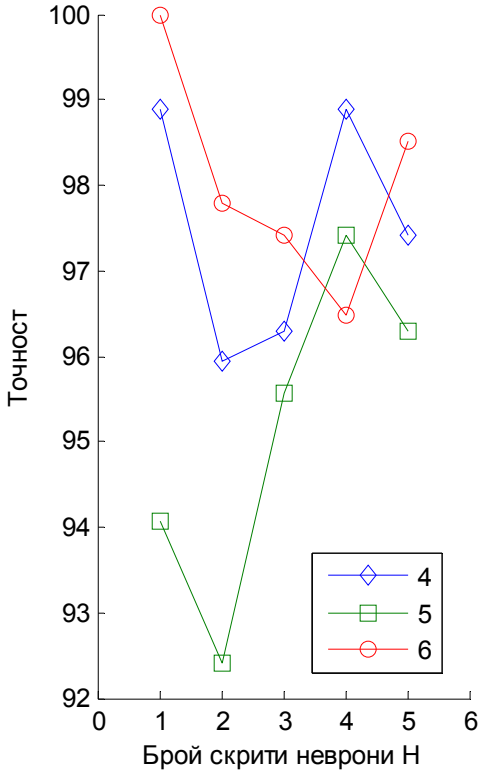

Участник #1
Точност
Брой скрити неврони H
1
2
3
4
5
6

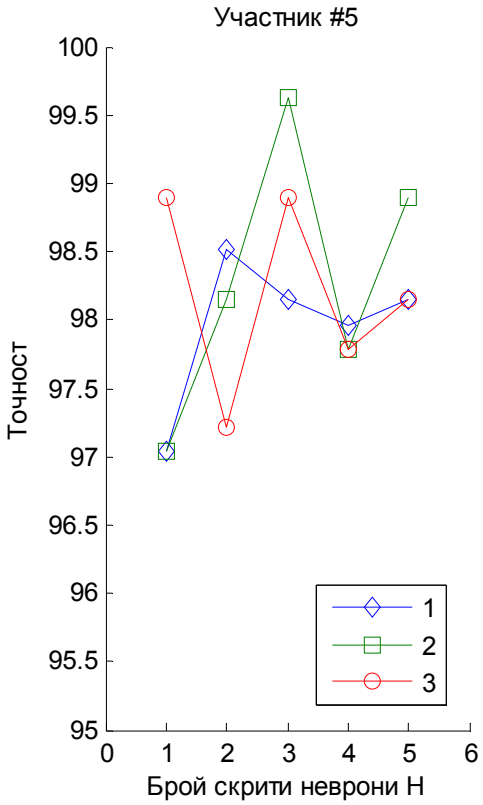

Участник #2
Точност
Брой скрити неврони H
1
2
3
4
5
6

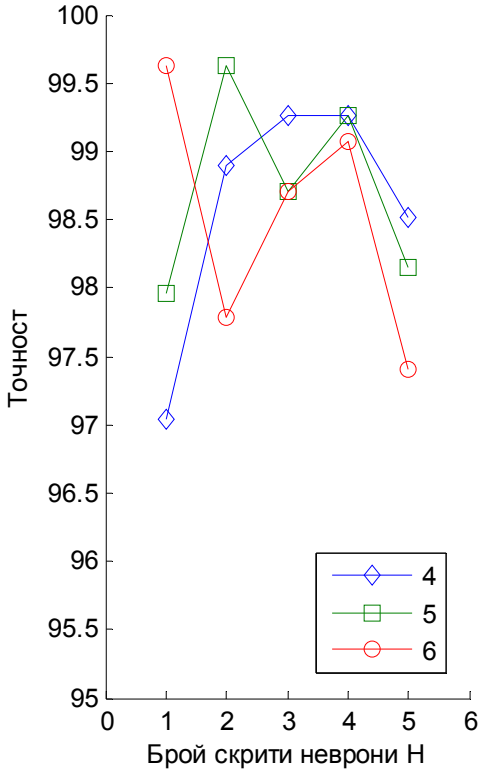

Участник #3
Точност
Брой скрити неврони H
1
2
3

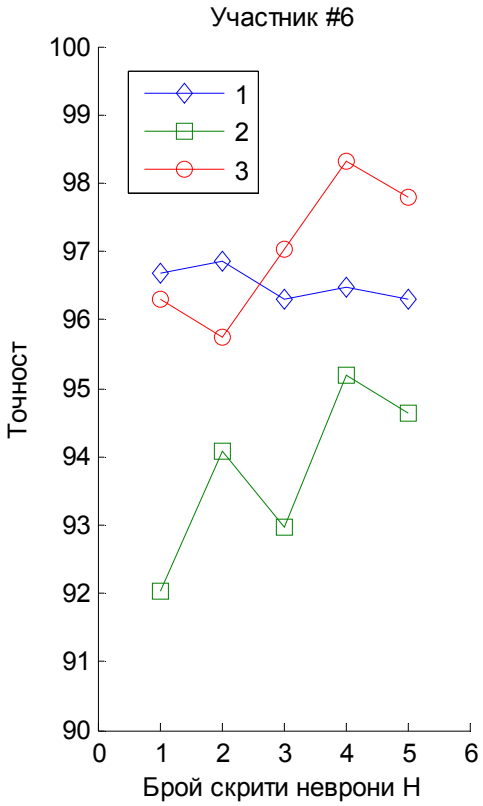

Точност
Брой скрити неврони H
4
5
6

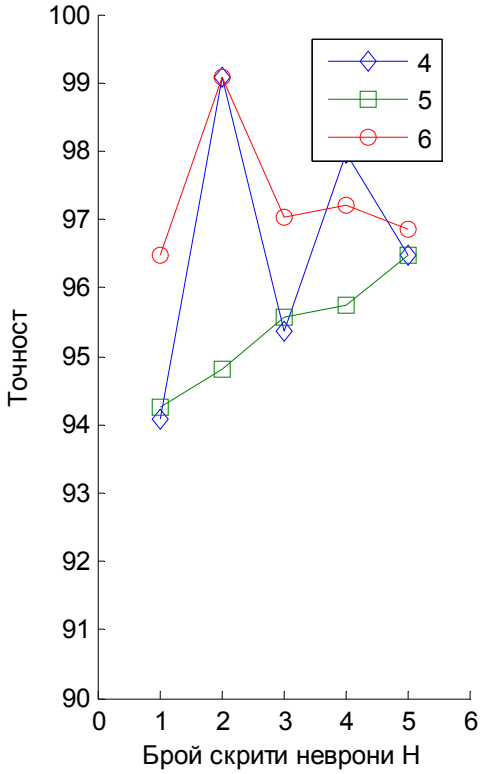

Участник #4
Точност
Брой скрити неврони H
1
2
3

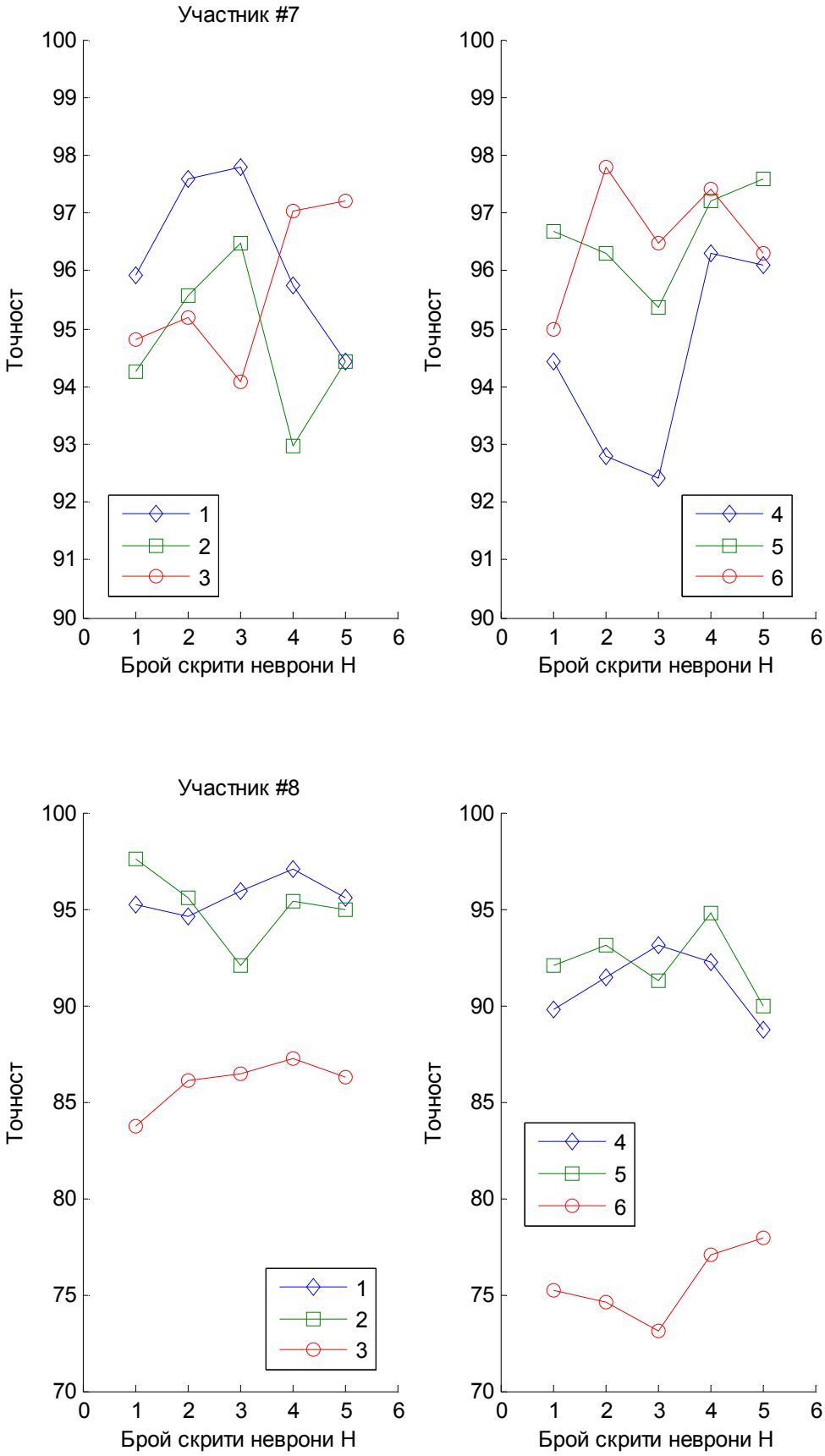

Точност
Брой скрити неврони H
4
5
6


Участник #5
Точност
Брой скрити неврони H
1
2
3


Точност
Брой скрити неврони H
4
5
6


Участник #6
Точност
Брой скрити неврони H
1
2
3


Точност
Брой скрити неврони H
4
5
6



Фигура 4.1 Оценка на точността на верификация с НМ

Проектирани са класификатори по *k*-най-близки съседи за всеки участник за стойности $k = 1$ и $k = 3$. Общият им брой е 12. Те се оценяват и сравняват чрез метода LOOCV.

В таблица 4.4 са отразени параметрите на избраните окончателни модели за двата вида класификатора за всеки участник. В случай на равни максимални стойности на точността за два или повече модела за даден участник, е избран този от тях, построен с по-малък брой признаци и / или брой скрити неврони (НМ) и / или брой съседи (*k*NN).

Таблица 4.4 Окончателни модели за всеки участник

Участник	**Избран модел (НМ)**	***H***	**Точност (НМ)**	**Избран модел (*k*NN)**	***k***	**Точност (*k*NN)**
1	2	3	~ 96%	2	1	~88%
2	6	3	~ 99,5%	1	1	~96%
3	6	3	~ 99,5%	5	1	~92%
4	5	1	~ 99,8%	1	1	~92%
5	5	1	~ 99,5%	5	1	~100%
6	5	2	~ 99%	5	1	~100%
7	5	2	~ 98%	2	1	~92%
8	3	1	~ 97%	4	1	~80%

Средната стойност на приближената оценка на точността за класификатор НМ за всички участници е равна на 97,5%, а тази на точността за класификатор *k*NN е равна на 92,5%.

След намирането на оптимални стойности на параметрите на класификатор по метода на крос валидация, той се обучава върху цялото множество от данни и бива използван на практика.

От Фигура 4.2, представяща точността на различните шест модела за разпознаване с НМ с и без избор на признаци може да се заключи, че изборът на признаци редуцира броя им, докато точността нараства или се запазва. Така, само част от признаците са необходими за построяването на надежден модел. Вторият експериментален извод се отнася до значимостта на вида използвани подправени подписи за точността на верификацията. От таблицата 4.4 с резултати се вижда, че за Случай “2” (умели и неумели фалшификати) се получава по-висока точност в сравнение със Случай “1” (само неумели фалшификати) за 6 от 8 участника.

4.4 Получени резултати върху база данни за подписи SUsig

Броят на събраните собствени подписи е различен за всеки потребител (Таблица 4.5).

Таблица 4.5 Брой собствени подписи на потребителите от SUsig

Участник	Брой подписи	Участник	Брой подписи	Участник	Брой подписи	Участник	Брой подписи
#1	8	**#24**	10	**#47**	10	**#70**	8
#2	8	**#25**	10	**#48**	8	**#71**	10
#3	8	**#26**	10	**#49**	8	**#72**	10
#4	10	**#27**	10	**#50**	10	**#73**	10
#5	10	**#28**	10	**#51**	10	**#74**	10
#6	10	**#29**	10	**#52**	8	**#75**	10
#7	8	**#30**	10	**#53**	10	**#76**	10
#8	8	**#31**	10	**#54**	10	**#77**	8
#9	10	**#32**	10	**#55**	8	**#78**	10
#10	10	**#33**	8	**#56**	10	**#79**	10
#11	10	**#34**	8	**#57**	10	**#80**	10
#12	10	**#35**	8	**#58**	10	**#81**	10
#13	10	**#36**	8	**#59**	10	**#82**	8
#14	8	**#37**	10	**#60**	10	**#83**	8
#15	8	**#38**	8	**#61**	10	**#84**	10
#16	8	**#39**	10	**#62**	10	**#85**	10
#17	8	**#40**	10	**#63**	10	**#86**	10
#18	10	**#41**	10	**#64**	8	**#87**	8
#19	8	**#42**	10	**#65**	10	**#88**	10
#20	8	**#43**	10	**#66**	10	**#89**	8
#21	8	**#44**	10	**#67**	10		
#22	8	**#45**	10	**#68**	10		
#23	10	**#46**	10	**#69**	8		

Намиране на общо подмножество от признаци

За базата данни за подписи SUsig не разполагаме със стойности на признаците A25 и A26 (средна стойност на наклона на писалката и средна стойност на азимут) и затова началният брой признаци е 24. За всеки потребител за всяка двойка признаци, е пресметната стойността на взаимната им корелация. В таблица 4.6 е отразен броя на срещанията (в случай че надвишава 22, т.е. се среща при повече от 25% от потребителите) на признаците, за които се наблюдава висока корелация за всичките 89 потребители. На базата на тази таблица, с прилагане на метода на корелационните плеяди [99], премахваме от по-нататъшно разглеждане признаците A3, A5, A7, A8, A9, A11, A14, A15, A18, A19, A20. Броят на признаците се редуцира с около 50% и тези, които остават са 13 на брой и са следните: A1, A2, A4, A6, A10, A12, A13, A16, A17, A21, A22, A23, A24.

Таблица 4.6 Брой срещания на двойки признаци с висока взаимна корелация - SUsig

Признак 1	Признак 2	Брой срещания
A3	A2	27
A5	A4	41
A7	A2	20
A7	A3	52
A9	A1	29
A10	A8	45
A10	A9	30
A13	A11	37
A14	A1	53
A15	A1	56

Признак 1	Признак 2	Брой срещания
A18	A8	30
A18	A10	25
A18	A11	28
A19	A1	47
A19	A9	52
A19	A10	35
A19	A14	34
A19	A15	36
A20	A11	25
A23	A5	34

Намиране на индивидуално подмножество от признаци за всеки потребител

Резултатите от избора на най-информативни признаци по метода на *Hocking* и *Leslie* (Раздел 2.3) за всеки от потребителите са обобщени в таблица 4.7. Вариант “1” се отнася до случая, при който подмножеството от индивидуални признаци се определят от 8 или 10 собствени подписа (Таблица 4.6) и 10 неумели фалшификати, а Вариант “2” – от 8 или 10 собствени и 10 умели фалшификати.

Таблица 4.7 Редуцирани признаци за всеки потребител от SUsig

Участник	Вар.	A1	A2	A4	A6	A10	A12	A13	A16	A17	A21	A22	A23	A24	Ср	p
	"1"	+		+	+	+		+	+			+	+	+	5,35	9
#1	"2"	+			+		+	+	+		+	+	+	+	5,25	9
	"1"	+			+	+	+	+		+	+	+		+	7,77	9
#2	"2"		+	+	+		+		+	+	+	+		+	5,06	9
	"1"		+	+	+	+	+	+	+	+				+	7,65	9
#3	"2"	+	+		+			+	+		+			+	6,39	7
	"1"					+			+	+		+		+	4,8	5
#4	"2"				+			+	+			+		+	2,94	5
	"1"	+	+	+		+	+				+			+	4,94	7
#5	"2"	+	+	+		+		+			+	+	+	+	5,22	9
	"1"	+	+	+	+	+	+			+				+	8	9
#6	"2"	+	+	+	+		+	+		+	+			+	6,42	9
	"1"	+	+	+	+		+			+			+	+	5,24	8
#7	"2"	+	+	+	+	+			+	+		+		+	5,01	9
	"1"		+	+	+			+			+			+	2,22	6
#8	"2"	+	+	+	+	+	+				+	+		+	5,23	9
	"1"			+	+		+		+			+	+	+	4,47	7
#9	"2"		+	+	+		+		+	+		+	+	+	5,28	9
#10	"1"						+		+	+				+	2,03	4

	"2"	+		+	+	+	+				+	+	+	+	6,02	9
	"1"	+	+				+		+	+				+	3,1	6
#11	"2"	+	+	+	+			+	+		+	+		+	5,75	9
	"1"	+		+				+	+	+	+	+	+	+	5,57	9
#12	"2"			+						+				+	0,71	3
	"1"		+	+		+	+					+		+	5,39	6
#13	"2"	+	+	+	+	+		+			+	+		+	5,05	9
	"1"			+				+	+			+	+	+	3,24	6
#14	"2"	+			+		+	+		+	+		+	+	5,93	8
	"1"	+	+	+	+	+	+			+		+		+	5,39	9
#15	"2"		+	+	+	+	+	+			+	+		+	5,13	9
	"1"	+			+	+	+	+	+		+	+		+	5,01	9
#16	"2"	+				+		+	+	+	+	+	+	+	5,22	9
	"1"				+					+				+	2,96	3
#17	"2"	+	+		+		+		+	+	+		+	+	5,12	9
	"1"	+	+	+	+	+			+		+		+	+	5,29	9
#18	"2"			+		+			+	+		+		+	5,74	6
	"1"	+	+	+	+		+		+		+		+	+	7,11	9
#19	"2"	+		+	+		+	+		+		+	+	+	5,38	9
	"1"		+	+	+	+	+			+		+	+	+	5,48	9
#20	"2"				+			+			+		+	+	4,65	5
	"1"						+	+		+		+		+	4,94	5
#21	"2"		+	+	+			+	+	+	+	+		+	5,68	9
	"1"	+	+		+				+				+	+	3,75	6
#22	"2"	+		+	+	+		+	+		+	+		+	5,14	9
	"1"	+	+	+	+	+	+		+	+				+	5,25	9
#23	"2"	+		+			+	+	+		+	+	+	+	5,12	9
	"1"	+	+	+	+			+	+			+	+	+	6,71	9
#24	"2"	+	+		+			+				+		+	2,34	6
	"1"		+	+		+	+	+	+		+	+		+	5,04	9
#25	"2"	+	+	+	+			+	+		+		+	+	5,05	9
	"1"		+		+	+			+	+	+	+		+	6,84	8
#26	"2"		+	+		+				+			+	+	3,72	6
	"1"			+	+	+	+		+					+	3,47	6
#27	"2"		+	+	+		+	+	+		+	+		+	5,28	9
	"1"	+		+	+	+	+		+		+		+	+	8,41	9
#28	"2"	+	+	+								+		+	1,55	5
	"1"	+	+	+	+	+		+				+	+	+	12,15	9
#29	"2"	+	+	+		+	+	+					+	+	7,52	8
	"1"	+		+	+	+	+		+		+	+		+	5,27	9
#30	"2"				+				+			+		+	1,18	4
	"1"	+	+		+		+		+	+	+	+		+	8,81	9
#31	"2"		+	+	+		+	+	+		+			+	5,7	8
	"1"	+	+		+	+	+			+		+	+	+	8,38	9
#32	"2"	+	+	+			+		+	+	+	+		+	5,21	9
	"1"		+		+			+			+		+	+	4,49	6
#33	"2"	+			+	+			+	+	+	+	+	+	5,71	9

	"1"	+		+	+	+			+	+		+	+	+	5,39	9
#34	"2"	+		+	+	+				+	+	+	+	+	5,24	9
	"1"	+	+		+								+	+	4,16	5
#35	"2"	+			+						+	+		+	2,62	5
	"1"	+	+	+	+	+		+				+	+	+	5,2	9
#36	"2"	+		+		+	+	+	+		+	+		+	5,1	9
	"1"		+	+		+		+	+			+	+	+	7,43	8
#37	"2"	+	+			+		+		+	+	+		+	7,58	8
	"1"	+							+		+		+	+	4,38	5
#38	"2"	+	+		+		+		+		+	+	+	+	5,38	9
	"1"		+	+	+	+	+	+				+	+	+	5,13	9
#39	"2"	+	+	+		+					+			+	5,21	6
	"1"			+			+				+		+	+	4,67	5
#40	"2"		+	+	+					+				+	2,17	5
	"1"	+	+			+	+		+	+	+		+	+	7,48	9
#41	"2"	+		+	+	+			+	+		+	+	+	5,11	9
	"1"						+	+			+	+		+	2,5	5
#42	"2"	+		+		+			+					+	2,73	5
	"1"	+	+	+	+	+	+	+				+		+	5,04	9
#43	"2"	+		+			+				+		+	+	4,5	6
	"1"		+		+				+	+		+	+	+	3,68	7
#44	"2"	+				+	+		+	+		+		+	6,06	7
	"1"		+	+		+		+	+	+	+	+		+	5,35	9
#45	"2"		+	+	+	+		+		+	+	+		+	5,09	9
	"1"	+		+	+			+					+	+	4,72	6
#46	"2"	+	+	+	+			+		+	+	+		+	5,16	9
	"1"	+	+				+			+				+	1,84	5
#47	"2"	+	+	+	+	+				+		+	+	+	5,14	9
	"1"		+	+	+	+	+				+	+	+	+	5,18	9
#48	"2"	+	+		+	+	+	+			+	+		+	5,36	9
	"1"		+	+		+						+		+	3,11	5
#49	"2"	+	+	+	+	+			+		+		+	+	5,67	9
	"1"		+	+	+	+	+		+		+	+		+	7,57	9
#50	"2"			+					+		+	+		+	2,65	5
	"1"		+	+	+	+	+		+	+	+			+	5,22	9
#51	"2"	+		+	+	+	+	+			+		+	+	6,35	9
	"1"	+	+	+	+	+		+	+		+			+	5,16	9
#52	"2"	+	+		+		+		+		+	+	+	+	5,08	9
	"1"	+	+	+		+			+	+	+		+	+	5,38	9
#53	"2"			+	+	+		+	+					+	3,93	6
	"1"		+			+		+	+		+		+	+	5,92	7
#54	"2"	+		+	+	+	+		+			+	+	+	8,69	9
	"1"	+					+						+	+	2,3	4
#55	"2"		+	+	+	+	+		+		+	+		+	5,21	9
	"1"	+		+		+	+	+	+		+		+	+	8,13	9
#56	"2"	+	+		+		+	+			+	+		+	7,58	8
#57	"1"		+	+			+	+	+	+		+	+	+	8,19	9

	"2"	+		+			+		+	+	+	+	+	+	5,73	9
	"1"		+	+		+	+	+		+	+		+	+	5,98	9
#58	"2"	+	+		+			+	+	+	+		+	+	8,96	9
	"1"	+		+	+	+	+			+		+	+	+	6,46	9
#59	"2"		+	+		+		+		+	+	+	+	+	5,44	9
	"1"		+	+			+	+				+		+	3,93	6
#60	"2"	+	+		+	+		+			+			+	6,03	7
	"1"		+	+	+	+		+				+	+	+	7,52	8
#61	"2"			+		+			+	+				+	3,79	5
	"1"	+	+			+	+	+		+		+	+	+	8,31	9
#62	"2"		+		+	+	+		+	+	+		+	+	5,45	9
	"1"		+	+	+					+			+	+	3,56	6
#63	"2"			+								+		+	0,69	3
	"1"				+			+			+		+	+	4,72	5
#64	"2"	+		+	+		+	+	+		+	+		+	5,03	9
	"1"	+			+	+	+	+	+			+	+	+	5,1	9
#65	"2"		+	+			+		+			+	+	+	6,89	7
	"1"	+	+	+	+	+	+			+	+			+	8,34	9
#66	"2"		+	+	+			+	+	+	+		+	+	5,22	9
	"1"		+		+		+		+	+			+	+	5,67	7
#67	"2"	+			+		+	+				+		+	3,61	6
	"1"		+		+				+				+	+	4,38	5
#68	"2"				+		+					+	+	+	4,95	5
	"1"	+		+		+		+	+		+		+	+	5,74	8
#69	"2"		+	+	+	+		+	+	+	+			+	5,27	9
	"1"	+	+	+	+	+	+		+			+		+	8,81	9
#70	"2"		+		+	+	+	+	+		+		+	+	5,1	9
	"1"							+					+	+	2,76	3
#71	"2"			+		+		+					+	+	3,5	5
	"1"		+								+			+	2,6	3
#72	"2"		+		+			+	+					+	4,66	5
	"1"	+	+									+		+	1,28	4
#73	"2"		+	+		+	+		+	+		+		+	5,89	8
	"1"	+			+	+	+	+		+	+		+	+	5,09	9
#74	"2"	+		+		+				+				+	3,68	5
	"1"	+	+			+	+	+	+					+	4,29	7
#75	"2"	+			+				+	+				+	4,15	5
	"1"	+	+	+	+			+	+			+	+	+	8,02	9
#76	"2"			+		+		+				+		+	4,36	5
	"1"			+		+		+	+					+	2,67	5
#77	"2"		+	+		+	+		+	+	+		+	+	5,02	9
	"1"	+	+				+		+	+	+	+	+	+	6,63	9
#78	"2"					+			+				+	+	1,1	4
	"1"	+		+		+	+	+			+	+	+	+	8,06	9
#79	"2"	+	+	+	+		+			+		+	+	+	8,41	9
	"1"			+						+			+	+	3,34	4
#80	"2"			+	+	+			+	+				+	5,78	6

	"1"	+				+				+	+		+	+	4,26	6
#81	"2"			+						+	+	+	+	+	4,28	6
	"1"	+				+		+					+	+	3,66	5
#82	"2"				+			+			+			+	2,89	4
	"1"	+	+		+	+		+	+	+	+			+	5,33	9
#83	"2"	+		+	+		+	+		+	+		+	+	5,26	9
	"1"			+					+	+	+		+	+	4,44	6
#84	"2"		+				+		+	+	+	+	+	+	6,81	8
	"1"						+			+	+		+	+	3,89	5
#85	"2"	+		+					+	+				+	4,23	5
	"1"						+					+		+	1,01	3
#86	"2"		+	+	+	+		+		+	+	+		+	5,68	9
	"1"		+	+	+		+		+			+		+	6,64	7
#87	"2"	+	+		+	+		+	+			+	+	+	5,96	9
	"1"		+						+	+				+	3,36	4
#88	"2"			+	+	+			+			+		+	5,26	6
	"1"	+			+					+			+	+	3,65	5
#89	"2"	+		+			+	+	+	+		+	+	+	5,61	9

В Таблица 4.8 и за двата варианта са посочени брой на редуцираните признаци и съответния брой участници.

Таблица 4.8 Брой срещания на редуцирани признаци

Вар.	**Брой на редуцирани признаци**	**Брой участници**
“1”	9	42
“1”	8	5
“1”	7	7
“1”	6	12
“1”	5	14
“1”	4	5
“1”	3	4

Вар.	**Брой на редуцирани признаци**	**Брой участници**
“2”	9	48
“2”	8	9
“2”	7	4
“2”	6	10
“2”	5	15
“2”	4	3
“2”	3	0

Вижда се, че е постигнато значително редуциране на броя на признаците, понеже първоначалния им брой (равен на 13) се редуцира до 9 за около половината участници и при двата варианта, а за 30% от участниците (26 души) и при двата варианта се намалява до 5 или 6 признака. При малък брой участници признаците достигат брой, по-малък от 5 (за Вариант “1” – 9 души, а за Вариант “2” – само 3). Тези резултати показват, че първоначалният брой признаци се редуцира значително и е по-малък при използване на неумели фалшификати за отрицателни примери (Вариант “1”).

Параметри на построените модели

Таблица 4.9 съдържа обобщена информация за всеки един от построените модели (с номера от 1 до 6) за базата от данни за подписи SUsig.

Таблица 4.9 Параметри на моделите за базата от данни SUsig

<table>
<tr><th># на модел</th><th>Брой признаци</th><th>Брой собствени подписи</th><th colspan="2">Брой подправени подписи</th><th>Брой скрити неврони H</th><th>Брой съседи k</th></tr>
<tr><td>1</td><td>Общи признаци</td><td rowspan="6">8 или 10</td><td rowspan="3">15 неумели</td><td rowspan="3">Случай “1”</td><td rowspan="6">от 1 до 5</td><td rowspan="6">1 или 3</td></tr>
<tr><td>2</td><td>Вариант “1”</td></tr>
<tr><td>3</td><td>Вариант “2”</td></tr>
<tr><td>4</td><td>Вариант “2”</td><td rowspan="3">9 неумели и 6 умели</td><td rowspan="3">Случай “2”</td></tr>
<tr><td>5</td><td>Вариант “1”</td></tr>
<tr><td>6</td><td>Общи признаци</td></tr>
</table>

Построените модели се различават по вида на отрицателни примери за обучение и по броя на признаците.

Избор на параметри на класификатори НМ и *k*NN чрез крос валидация

За всеки участник се извършва оценка на точността за всяка от тридесетте архитектури на НМ (построени по 6 модела, като за всеки модел стойността на H варира от 1 до 5) по метода на 10-кратната крос валидация и оценка на точността на всеки от дванадесетте класификатори kNN (построени по 6 модела, като за всеки модел стойността на k е равна на 1 или на 3) по LOOCV, а за окончателен се избира този модел и съответната стойност на H или k, който демонстрира най-висока точност.

В Таблица 4.10 са посочени параметрите на така избраните модели за всеки участник. В случай на равни максимални стойности на точността за два или повече модела на НМ за даден участник, е избран този от тях, построен с по-малък брой признаци и/ или по-малък брой скрити неврони. В случай на равни максимални стойности на точността за два или повече модела на kNN за даден участник, е избран този от тях, построен за по-малък брой признаци и по-малка стойност на k. В колоните “Брой съседи k“ и “Брой скрити неврони H“ са посочени стойностите на k и H, за които се получава най-добър резултат. В колона “№ на избран модел“ е посочен номера на модела, за който е получен най-добър резултат.

Таблица 4.10 Окончателни модели за всеки участник от SUsig

Участник	№ на избран модел за НМ	Брой скрити неврони *H*	Точност крос валидация (%)	№ на избран модел за kNN	Брой съседи *k*	Точност LOOCV (%)
#1	1	5	100	2	1	100.00
#2	3	3	98.33	4	1	95.65
#3	5	5	93.7	5	3	86.96
#4	4	2	95.93	3	1	92.00
#5	2	5	98.89	2	1	96.00
#6	4	4	98.15	5	1	100.00
#7	3	4	97.41	5	1	95.65
#8	1	3	97.41	4	3	86.96
#9	5	2	98.89	4	1	96.00
#10	4	4	99.07	2	3	96.00
#11	6	2	98.52	3	1	92.00
#12	6	1	99.63	5	1	92.00
#13	1	1	93.33	5	1	96.00
#14	2	3	99.26	2	3	100.00
#15	2	4	99.26	2	3	100.00
#16	2	2	99.07	2	1	100.00
#17	3	1	97.78	3	1	100.00
#18	2	1	96.48	2	1	92.00
#19	5	4	99.63	5	1	100.00
#20	6	3	97.78	5	1	95.65
#21	1	5	99.26	1	1	100.00
#22	5	5	97.41	2	1	95.65
#23	1	3	90	4	1	80.00
#24	5	3	98.7	2	1	96.00
#25	2	2	97.78	4	1	100.00
#26	4	2	97.22	4	1	100.00
#27	2	4	97.78	5	1	96.00
#28	2	2	98.89	4	1	96.00
#29	2	4	99.63	4	1	96.00
#30	3	3	97.04	4	1	92.00
#31	6	3	99.44	4	1	100.00
#32	2	2	98.15	5	1	100.00
#33	4	4	98.15	2	1	91.30
#34	2	2	97.96	2	1	91.30
#35	6	4	97.04	4	1	91.30
#36	2	4	99.44	2	1	95.65
#37	2	5	98.33	5	1	92.00
#38	1	2	99.63	2	1	100.00
#39	4	3	98.52	1	1	100.00
#40	2	2	94.07	5	1	88.00
#41	1	5	98.89	3	1	100.00
#42	2	4	98.52	2	1	96.00
#43	4	2	99.26	3	1	96.00

#44	4	2	97.59	4	1	100.00
#45	3	5	96.3	5	1	92.00
#46	1	5	96.67	2	3	92.00
#47	3	2	99.26	2	1	100.00
#48	4	4	97.41	4	1	86.96
#49	2	2	100	3	1	95.65
#50	1	3	97.04	3	1	96.00
#51	1	3	97.96	2	1	96.00
#52	2	3	99.44	1	1	100.00
#53	3	4	97.59	2	3	96.00
#54	3	4	96.11	2	1	92.00
#55	5	3	99.07	5	1	100.00
#56	5	4	98.52	5	1	92.00
#57	6	5	99.44	3	1	100.00
#58	6	4	91.67	2	1	76.00
#59	2	4	99.63	5	1	100.00
#60	6	2	96.3	5	3	88.00
#61	2	4	98.33	5	1	100.00
#62	6	2	97.59	4	1	96.00
#63	6	2	99.44	5	1	100.00
#64	3	5	99.07	2	1	100.00
#65	2	5	98.52	3	1	100.00
#66	1	2	99.26	3	1	100.00
#67	3	3	99.26	2	1	100.00
#68	2	2	98.7	3	1	96.00
#69	3	4	98.33	3	1	100.00
#70	5	2	100	4	3	95.65
#71	3	5	95.56	4	1	96.00
#72	1	2	99.44	2	1	100.00
#73	2	4	100	2	1	100.00
#74	4	4	99.63	4	1	100.00
#75	2	5	97.22	2	1	96.00
#76	4	5	92.78	6	3	84.00
#77	5	3	95.93	5	1	95.65
#78	2	5	98.52	3	1	100.00
#79	2	4	95.19	2	1	92.00
#80	4	2	97.96	4	1	96.00
#81	6	5	97.96	5	1	100.00
#82	4	4	99.63	4	1	100.00
#83	3	4	100	3	1	100.00
#84	2	1	98.89	3	1	100.00
#85	2	2	97.78	2	1	100.00
#86	1	5	96.85	4	1	100.00
#87	4	1	96.48	1	1	100.00
#88	4	3	100	1	1	100.00
#89	4	3	99.63	4	1	100.00

Сравнение на точността на класификаторите НМ и *k*NN

Целта на сравнението е да се провери как избраният подход за построяване на класификатор (НМ) се съотнася към други широко използвани класификатори. При малки извадки, когато оптимални в статистически смисъл подходи, не са приложими, най-често се използват евристични подходи като k – най-близки съседи. Следва да се отбележи, че при голям брой наблюдения и голямо k този подход е квази оптимален, т.е. води до получаване на класификатор, близък до оптималния Бейсов класификатор. В Таблици 4.10 – 4.12 са дадени резултатите от проведеното изследване.

Средната стойност на точността на НМ, оценена за всички участници от SUsig по избраните за тях модели чрез 10-кратна крос-валидация, е равна на 97.95%. Средната стойност на точността на *k*NN, оценена за всички участници от SUsig по избраните за тях модели чрез LOOCV, е равна на 96.13%. Тези резултати показват предимство за невронната мрежа. За отбелязване е, че най-добри резултати за *k*NN се получават при $k = 1$ за 79 потребители или 89% от случаите. За изчислените средни стойности и средноквадратични отклонения е изчислена стойността на t-статистиката $t = 3.29$. При 176 степени на свобода тази стойност е значима с вероятност 0.99.

Зависимост на точността на верификация от подмножеството признаци за разпознаване и вида фалшиви подписи за обучение

Изследването има за цел (1) да се оцени ефекта от редуцирането на броя на признаците върху точността на класификацията (2) да се изследва зависимостта на точността на класификация от вида фалшиви подписи, използвани за обучение на класификаторите.

Таблица 4.11 Таблица със средни стойности на оценената точност по най-добрите модели на НМ

	Общи признаци	**Вариант 1**	**Вариант 2**	**Средна точност**	**Брой случаи**
Случай 1	**97.36** *(Модел 1, 13 случая)*	**98.36** *(Модел 2, 27 случая)*	**97.85** *(Модел 3, 13 случая)*	**97.99**	**53**
Случай 2	**97.71** *(Модел 6, 11 случая)*	**97.98** *(Модел 5, 9 случая)*	**97.96** *(Модел 4, 16 случая)*	**97.89**	**36**
Средна точност	**97.52**	**98.27**	**97.91**	**-**	**-**
Брой случаи	**24**	**36**	**29**	**-**	**89**

За целта изчисляваме средните стойности на оценената точност по най-добрите модели за НМ (Таблица 4.11) и *k*NN (Таблица 4.12), средната точност по Общи признаци, по признаци от Вариант "1" и по признаци от Вариант "2" и средната точност по Случай "1" и по Случай "2".

Таблица 4.12 Таблица със средни стойности на оценената точност по най-добрите модели на kNN

	Общи признаци	**Вариант 1**	**Вариант 2**	**Средна точност**	**Брой случаи**
Случай 1	**100** *(Модел 1, 5 случая)*	**95.92** *(Модел 2, 27 случая)*	**97.84** *(Модел 3, 15 случая)*	**96.97**	**47**
Случай 2	**84** *(Модел 6, 1 случай)*	**95.5** *(Модел 5, 20 случая)*	**95.45** *(Модел 4, 21 случая)*	**95.20**	**42**
Средна точност	**97.33**	**95.74**	**96.45**	**-**	**-**
Брой случаи	**6**	**47**	**36**	**-**	**89**

Данните от Таблица 4.11 показват по-голяма точност на НМ при използване на индивидуалните признаци (Вариант "1" и Вариант "2") спрямо точността по общи признаци като се наблюдава превес на точността по признаци от Вариант "1". Точността по Вариант "1", Случай "1" (неумели фалшификати за обучение) надвишава точността по Вариант "2", Случай "2" (умели и неумели фалшификати за обучение).

При прилагане на класификатор *k*NN Таблица (4.12) точността по общите признаци надвишава тази по индивидуалните. Точността по Вариант "2", Случай "1" (неумели фалшификати за обучение) надвишава точността по Вариант "2", Случай "2" (умели и неумели фалшификати за обучение).

Сравнението на средната точност между НМ и *k*NN по общи и индивидуални признаци показва превъзходство на НМ.

Броят на избраните модели, обучени с неумели фалшификати (Случай "1"), се среща при повече от половината участници. Той е равен на 53 (при 60 % от участниците) за НМ и равен на 47 (при 53 % от участниците) за *k*NN. Броят на избраните модели, построени по признаци от Вариант "1" е най-голям за НМ (за 36 участника), както и за *k*NN (за 47 участника).

Построяване на класификатори НМ и *k*NN по избраните модели за всеки потребител

В Таблица 4.13 са показани резултатите от оценката на горните класификатори върху едни и същи извадки от 830 собствени и 890 фалшиви подписи. Около 70% от подписите се използват за проектиране на класификатора, а останалите 30% за тестване. При невронните мрежи за обучение са използвани по 3 или 4 собствени подписа и 7 фалшиви, а валидацията е извършена върху 2 или 3 собствени и 3 фалшиви подписа за всеки потребител. Тестването е извършено върху 3 собствени и 5 фалшиви подписа. При 9-кратна инициализация е избрана тази невронна мрежа, за която е получена най-добра оценка. За *k*NN са използвани 5 или 7 собствени подписа и 10 фалшиви, а за тестване – 3 собствени и 5 фалшиви. Разликата в броя използвани собствени подписи за обучение се дължи на различния брой събрани подписи от всеки (Таблица 4.6). В Таблица 4.13 за съответните модели за всеки участник са посочени стойностите на праг на НМ, номера на инициализация на теглата, стойности на FAR, FRR и точността по НМ и *k*NN.

Таблица 4.13 Оценка на качеството на класификация с НМ и kNN за всички участници

Участник	Праг (НМ)	№ на инициа лизация	FAR (НМ)	FRR (НМ)	Точност по НМ	FAR (*k*NN)	FRR (kNN)	Точност по kNN
#1	0.82	7	0	0	100	0	0	100
#2	0.82	6	0	0	100	20	0	87.5
#3	0.77	4	0	0	100	0	0	100
#4	0.83	5	0	0	100	0	0	100
#5	0.92	3	0	0	100	0	0	100
#6	0.92	2	0	0	100	0	0	100
#7	0.82	2	0	0	100	0	0	100
#8	0.81	4	0	0	100	20	33.33	75
#9	0.92	1	0	0	100	0	0	100
#10	0.92	7	0	0	100	0	0	100
#11	0.91	3	0	0	100	0	0	100
#12	0.87	4	0	0	100	0	0	100
#13	0.92	2	0	0	100	0	66.67	75
#14	0.82	8	0	0	100	0	0	100
#15	0.81	9	0	0	100	40	0	75
#16	0.82	7	0	0	100	0	33.33	87.5
#17	0.77	4	0	0	100	0	33.33	87.5
#18	0.92	6	0	0	100	0	33.33	87.5
#19	0.82	9	0	0	100	0	0	100
#20	0.8	9	0	0	100	0	0	100
#21	0.82	9	0	0	100	20	0	87.5
#22	0.86	2	40	0	75	0	33.33	87.5

#23	0.9	1	0	0	100	40	0	75
#24	0.92	2	0	0	100	0	33.33	87.5
#25	0.92	2	0	0	100	20	66.67	62.5
#26	0.92	3	0	0	100	40	0	75
#27	0.91	7	0	0	100	20	33.33	75
#28	0.92	1	0	0	100	0	33.33	87.5
#29	0.91	3	0	0	100	20	0	87.5
#30	0.82	1	40	0	75	60	66.67	37.5
#31	0.92	7	0	0	100	0	0	100
#32	0.8	6	0	0	100	0	0	100
#33	0.8	1	0	0	100	40	33.33	62.5
#34	0.82	8	0	0	100	20	0	87.5
#35	0.82	6	0	0	100	0	33.33	87.5
#36	0.82	5	0	0	100	0	0	100
#37	0.85	5	40	0	75	0	66.67	75
#38	0.82	4	0	0	100	0	0	100
#39	0.92	4	0	0	100	0	0	100
#40	0.8	2	0	0	100	0	66.67	75
#41	0.92	8	0	0	100	0	0	100
#42	0.91	8	0	0	100	0	0	100
#43	0.92	9	0	0	100	0	33.33	87.5
#44	0.91	9	0	0	100	20	0	87.5
#45	0.87	8	0	0	100	0	33.33	87.5
#46	0.91	8	0	0	100	0	0	100
#47	0.92	4	0	0	100	0	0	100
#48	0.82	1	0	0	100	40	33.33	62.5
#49	0.81	1	0	0	100	0	0	100
#50	0.89	9	0	0	100	0	0	100
#51	0.91	1	0	0	100	20	0	87.5
#52	0.82	3	0	0	100	0	0	100
#53	0.91	2	0	0	100	0	33.33	87.5
#54	0.83	2	0	0	100	0	33.33	87.5
#55	0.82	3	0	0	100	0	33.33	87.5
#56	0.88	5	0	0	100	0	33.33	87.5
#57	0.92	3	0	0	100	0	0	100
#58	0.89	5	40	0	75	40	66.67	50
#59	0.92	4	0	0	100	0	66.67	75
#60	0.87	2	0	0	100	40	0	75
#61	0.91	7	0	0	100	0	33.33	87.5
#62	0.92	1	0	0	100	20	33.33	75
#63	0.9	5	0	0	100	0	0	100
#64	0.81	4	0	0	100	0	0	100
#65	0.85	5	0	0	100	0	0	100
#66	0.92	3	0	0	100	60	0	62.5
#67	0.91	7	0	0	100	0	0	100
#68	0.91	8	0	0	100	0	33.33	87.5

#69	0.82	9	0	0	100	0	0	100
#70	0.82	2	0	0	100	0	0	100
#71	0.75	5	0	0	100	0	0	100
#72	0.92	1	0	0	100	0	0	100
#73	0.92	3	0	0	100	20	33.33	75
#74	0.92	1	0	0	100	0	33.33	87.5
#75	0.87	4	0	0	100	0	0	100
#76	0.9	6	0	0	100	0	0	100
#77	0.82	2	0	0	100	0	0	100
#78	0.88	9	0	0	100	0	0	100
#79	0.86	9	0	0	100	20	0	87.5
#80	0.65	3	40	0	75	20	33.33	75
#81	0.92	4	0	0	100	0	33.33	87.5
#82	0.82	8	0	0	100	0	33.33	87.5
#83	0.81	5	0	0	100	0	0	100
#84	0.9	8	0	0	100	0	0	100
#85	0.9	6	0	0	100	0	0	100
#86	0.92	7	0	0	100	20	0	87.5
#87	0.78	2	0	0	100	0	0	100
#88	0.92	9	0	0	100	20	0	87.5
#89	0.82	3	40	0	87.5	20	0	87.5

Получените стойности за средната точност, лъжливо положителните и лъжливо отрицателните отговори за невронната мрежа са съответно 98.46 %, 2.70 % и 0 %, а за kNN – 89.47 %, 8.09 % и 14.61 %. Те показват превъзходство на невронната мрежа спрямо *k*NN класификатора. Изчислената стойност на t-статистиката t = 5.98 показва значима разлика с вероятност 0.99.

Сравнение на резултатите с тези от други автори

Върху същата база данни за подписи SUsig е проектиран класификатор от авторите *Yanikoglu* и *Kholmatov* [6], който е оценен с 1.64% *FRR* и 1.28% *FAR*.

В [51] са тествани различни архитектури на невронни мрежи за верификация на подпис. Най-добрият получен експериментален резултат е 2% FAR и 1.3% FRR за многослоен перцептрон с един скрит слой. Във всичките експерименти са използвани по 5 подписа от всеки потребител за обучение и 41 глобални признаци на входа.

В [106] е представен класификатор *k*NN за оф-лайн подписи. Стойността на точността е оценена чрез крос валидация и е раван на 97.47%. В [107] и [108] са представени класификатори *k*NN за оф-лайн подписи, за които е постигната точност съответно от 70% и 78%.

Така представените в дисертацията класификатори имат оценки сходни с тези, които се срещат в литературата с използване на същата база данни за подписи, както и със

същите по вид класификатори.

4.5 Резултати в Глава 4

От проведените експерименти за избор на най-информативно подмножество от признаци от собствената база от подписи и базата SUsig, както и за точността на верификацията, могат да се направят следните изводи:

1. Последователното използване на метода на корелационните плеяди и регресионния подход, основан на *Cp* критерия на Mallows, води до съществено намаляване на броя на необходимите признаци (приблизително два пъти).
2. За всеки от участниците в експериментите съществува специфично подмножество от признаци, което описва най-добре съвкупността от неговите подписи или с други думи, не съществува подмножество от признаци, което еднакво добре да разграничава всеки от участниците. За някои от участниците индивидуалното подмножество се състои от 3-5 признака.
3. По-значителна редукция в броя на признаците се получава при използване на неумели фалшификати (Вариант “1”) като отрицателни примери.
4. Редукцията на броя на признаците при невронните мрежи води до повишаване на точността на верификация, докато при *k*NN се наблюдава намаляване на точността.
5. Обучените с неумели фалшификати класификатори (Случай ”1”) водят до по-добри резултати в сравнение с обучените с умели и неумели фалшификати (Случай “2”).

Тези изводи водят до следните препоръки за прилагане на изложения в дисертационната работа комбиниран метод за верификация по подпис:

1) да се редуцира броя на признаците с последователно прилагане на метода на корелационните плеяди и *Cp* критерия на Mallows;

2) Да се обучат класификатори по НМ с използване на неумели фалшификати и да се избере най-точния от тях за всеки конкретен участник.

Глава 5. Софтуерно приложение

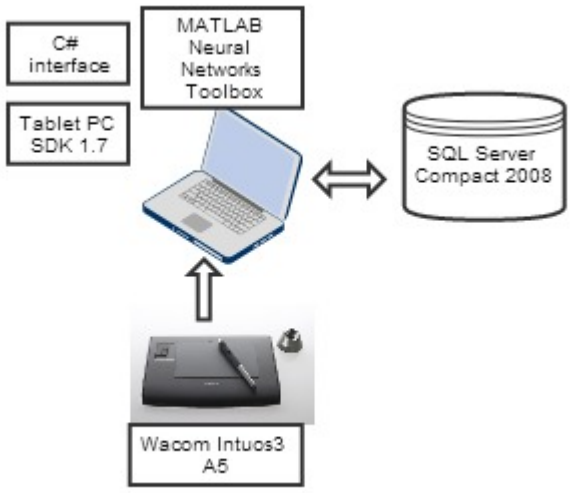
Една от целите на дисертационния труд е разработването на система за разпознаване по подписи. В тази глава са описани използваните за създаването й софтуер и хардуер, изграждащите я модули и екраните на проектираното софтуерно приложение. Представени са и събраните за изследването бази от данни, както и протокола за събиране на подписи.

5.1 Среда за разработка

Софтуерното приложение е създадено с помощта на средата за разработка на приложения *Microsoft Visual Studio 2008 Express Edition* [58] с програмен език *C#*. Събирането на подписите става възможно благодарение на пакета за разработка на приложения *Tablet PC SDK 1.7* [60], който е създаден от *Microsoft,* за да улесни създаването на приложения за работа с технологията на цифрово мастило (*digital ink*). Образците на подписи се събират с помощта на графичен таблет *Wacom Intuos3 A5 PTZ-630* [61] с размер на работната площ 152.4 x 210.6 mm. Този таблет поддържа 1024 нива на натиск, има разрешаваща способност от 5080 линии на инч и скорост на предаване на данните от 200 точки в секунда, която от своя страна определя интервала от време (в случая 0.05 секунди) между регистрирането на измерванията в две последователни точки. Има два вида писалки, с които може да работи конкретния графичен таблет. Първата е обикновена, т.нар. *grip pen* и при писане с нея не остават следи, за разлика от втората, т.нар. *inking pen*, с която може да се пише едновременно и върху лист хартия, поставен върху таблета. “Суровите” данни за подписите, както и обработените данни (според трансформациите, описани в Раздел 2.1 Предварителна обработка на потока от данни за подписи) се съхраняват в СУБД *SQL Server Compact Edition 2008* [59].

За експериментирането с различни топологии на невронни мрежи при търсенето на подходящ класификатор, е използван софтуерния продукт за научни изчисления MATLAB v.7.11.0 [62], с неговия инструментариум за работа с невронни мрежи *Neural Networks Toolbox*.

Изброените дотук софтуерни продукти са инсталирани върху лаптоп и настолен компютър с операционна система Microsoft Vista Ultimate SP1. На Фигура 5.1 е изобразена средата за разработка, която демонстрира взаимовръзката между основните хардуерни и софтуерни компоненти.



Фигура 5.1 Среда за разработка

5.2 Бази от данни за подписи

Експериментите с класификатори за верификация по подпис са извършени върху собствена база данни за подписи, събрана за целта, както и върху публично достъпната база данни за подписи SUsig .

5.2.1 Събиране на подписи

За съжаление, липсва стандартна база данни от подписи за свободно ползване сред научните среди, която да улесни експериментирането със СРП. Това е и причината изследователите да създават бази данни от подписи, а това е свързано със загуба на значително време и ресурси. Тъй като обикновено се използват подписите на хора от близкото обкръжение (както е направено и тук), то не е възможно създаването на достатъчно представителна разнообразна база данни от подписи, а именно включваща участници от различни социални групи – пол, националност, ляво и дясноръки, образование и т.н. Участниците в експериментите са колеги от секция “ОСРО”- БАН с различна възраст и пол. Всеки от тях демонстрира индивидуален стил на подписване, така подписите на някои от участниците представляват неразбираеми щрихи, а на други четим и стегнат почерк. За да бъде една система за разпознаване по подпис достоверна е необходимо да се разглеждат всички събрани подписи. Това означава да не се премахват едни или други подписи, тъй като в една реално работеща система това е недопустимо. Това важно изискване е спазено.

Подписите се събират мобилно с помощта на лаптоп и графичен таблет *Wacom Intuos3 A5 PTZ-630* [61], който дава възможност за визуална обратна връзка и позволява писането както със стандартна писалка за таблет (*grip pen*), така и с писалка с мастило (*inking pen*), с която може да се пише по стандартен начин върху лист, поставен върху таблета като същевременно се получава и динамичната информация за всяка точка от траекторията на подписа. Тъкмо

тази информация се записва в създадената база данни от подписи.

Броят на потребителите е 8. Мобилността на лаптопа и графичният таблет позволи събирането на подписи в удобно място и време за участниците. Те изразиха писмено съгласие за това подписите им да бъдат използвани за научни цели. Протоколът за събиране на подписи е посочен по-долу. Целта на този протокол е описанието на процеса и детайлите по практическото събиране на образци на подписи, което се извършва с помощта на разработеното за целта софтуерно приложение.

Протокол за събиране на подписи

Протоколът за събиране на подписи (*acquisition protocol*) се състои от следните стъпки:

1) Участниците се разделят в две групи по 4 души, биват инструктирани за начина за събирането на образци на подписи и им се извършва демонстрация;
2) Последователно се дава възможност на всеки от участниците да свикне да работи с графичния таблет и да експериментира с него, като се изисква да се подписват с възможно най-много вариации на подписа, без изкуствени забавяния и накъсвания на подписа. Не се налагат ограничения върху местоположението на подписа върху таблета, както и за неговата ориентация;
3) Всеки участник в седнало положение полага десет броя собствени подписи върху отделен лист хартия, поставен върху таблета чрез писалката с мастило на графичния таблет;
4) В същото време останалите участници от групата, които ще фалшифицират неговите подписи го наблюдават и така добиват представа относно спецификата при подписването му – скорост, форма и т.н.;
5) След като всичките участници от групата положат подписите си, се провежда 10 минутна пауза, в която всеки от тях, разполагайки с подписите на останалите, ги разучава самостоятелно и експериментира върху подправянето им върху лист хартия;
6) При готовност от страна на участниците, те последователно полагат по 3 умели фалшифицирани подписа за останалите участници в групата, към която принадлежат.

Процесът по събиране на подписи се контролира от оператор, който изпълнява следните задачи:

1) Предварително обучава участниците за работа като им демонстрира етапите от протокола и им предоставя възможност да експериментират с графичния таблет. След като прецени, че даден участник е готов, се пристъпва към същинското събиране на образци на неговия подпис;
2) Работи със софтуерното приложение, в т.ч. записва нов потребител и изчиства екрана за събиране на подписи преди всяко полагане на подпис. За всеки конкретен образец съвместно с потребителя взима решение дали да се запамети в базата данни от подписи според това дали е положен успешно;
3) Направлява участниците в извършваните от тях действия;
4) Накрая унищожава листовете хартия с подписи.

Тъй като всеки от участниците имитира всеки от останалите в групата, то се постига разнообразие във фалшификатите, а това, че те са разделени в две групи им позволява да се концентрират върху имитирането на подписи от малък брой участници, което означава и по-добро качество на подправените подписи. Повечето от потребителите са любопитни относно начина, по който изглежда положеният от тях подпис. Възможността за визуален контрол върху написаното им вдъхва увереност, че контролират процеса при полагане на подпис с неприсъща за ежедневието им дигитална писалка.

И така, броят на събраните собствени подписи е 80 (по 10 подписа от потребител), а този на умелите фалшификати - 72 (по 9 подписа за потребител). Данните за всеки от подписите съдържат стойностите на координатите x и y, времеви отпечатък, ниво на натиск, наклон, азимут и индикатор за начало на щрих.

5.2.2 База данни за подписи SUsig

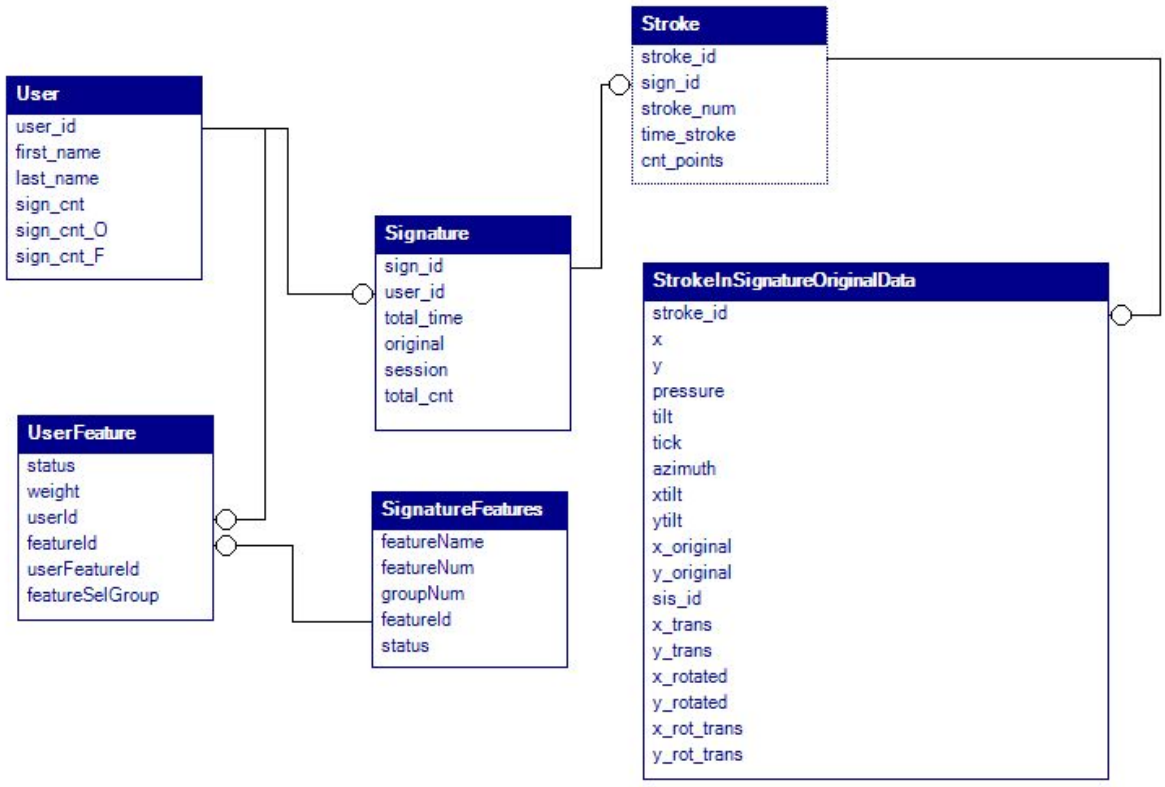
Базата данни за подписи SUsig [65] е предоставена от университета *Sabanci University* – Турция. Тя се състои от две части, съответно *Visual* и *Blind*. Първата е събрана 4 години преди *Blind* и съдържа подписи, събрани с помощта на таблет за подписи с вграден екран Interlink Elec. ePad, отчитащ 128 нива на натиск. За всеки потребител се събрани 20 собствени и 10 подправени подписа, като собствените подписи са събрани в две сесии. Преди да подправи подписа на даден участник, потребителят наблюдава анимация на съответния подпис и може да се упражни предварително във фалшифицирането му. Броят на участниците е 84. За събирането на подписите от втората част (*Blind*) е използван графичният таблет *Wacom Graphire2*, отчитащ 512 нива на натиск. С него са събрани по 8 или 10 собствени и 10 подправени подписа в една сесия за всеки участник. Преди да подправи подписа на даден участник, потребителят от *Visual* наблюдава анимация на съответния подпис и може да се

упражни предварително във фалшифицирането му. Предоставя му се и възможност да наблюдава и останалите собствени подписи за подправяне и така да се запознае с изменчивостта на дадения подпис. Броят на участниците е 89. За някои от потребителите броят на събраните собствени подписи е 8, а не 10 и поради това общият брой събрани подписи е 830. За всеки от подписите, в отделен текстов файл, се съдържа информация за координатите x и y, времеви отпечатък, ниво на натиск и индикатор за начало на щрих. Въз основа на предоставените характеристики на подписите, се пресмятат времетраенето и дължината (като брой точки) на всеки един от щрихите.

Използвани са подписите от частта *Blind* на базата данни от подписи SUsig, тъй като се наблюдава сходство между тях и събраните от нас подписи: те също са събрани в една сесия, с графичен таблет с подобни технически характериситки и с подобно качество на събраните фалшификати.

5.2.3 Диаграма на базата данни от подписи

Диаграмата на таблиците от базата данни е представена на Фигура 4.1.



Фигура 5.2 Диаграма на таблиците в БД

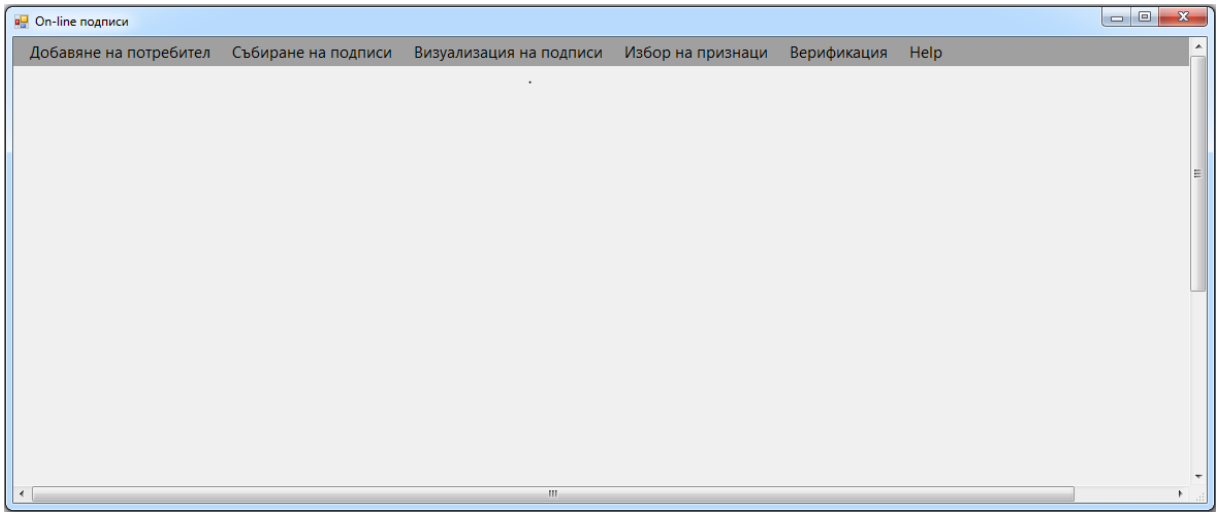
В таблицата *User* се съдържа информация за участниците, регистрирани в системата, а в *Signature* – за положените техни собствени и подправени подписи. Тъй като всеки подпис представлява съвкупност от един или повече щриха, е създадена и таблицата *Stroke*, в която

се записва пореден номер на щрих, времетраенето му и броя точки, както и това към кой от подписите в *Signature* се отнася. При полагането на всеки един от подписите (собствени и подправени), данните за неговите динамични характеристики (*x* и *y* координатите на върха на писалката, натиска, времевия отпечатък, азимута на писалката, съответния щрих и ъгъла на наклон на писалката за всяка точка от подписа) се записват в таблицата *StrokeInSignatureOriginalData.* След това се прилагат някои операции върху стойностите на координатите *x* и *y.* Извършваните операции са трансформация на координатната система, ротация на подписа (тъй като ориентацията на таблета и подписа не винаги съвпадат) и транслация на подписа в определена точка (вж. Раздел 2.1). Новополучените стойности на координатите *x* и *y* след всяка от посочените операции се записват в съответно поле от таблицата *StrokeInSignatureOriginalData*: *x_trans, y_trans, x_rotated, y_rotated, x_rot_trans, y_rot_trans*. В таблицата *SignatureFeatures* се съхраняват наименованията, статуса и пореден номер на използваните в системата признаци, докато в таблицата *UserFeature* са посочени избраните подмножества от признаци за всеки от потребителите.

Подписите от частта *Blind* на SUsig са добавени в създадената база данни за по-лесна обработка от софтуерното приложение за верификация по подпис.

5.3 Екрани

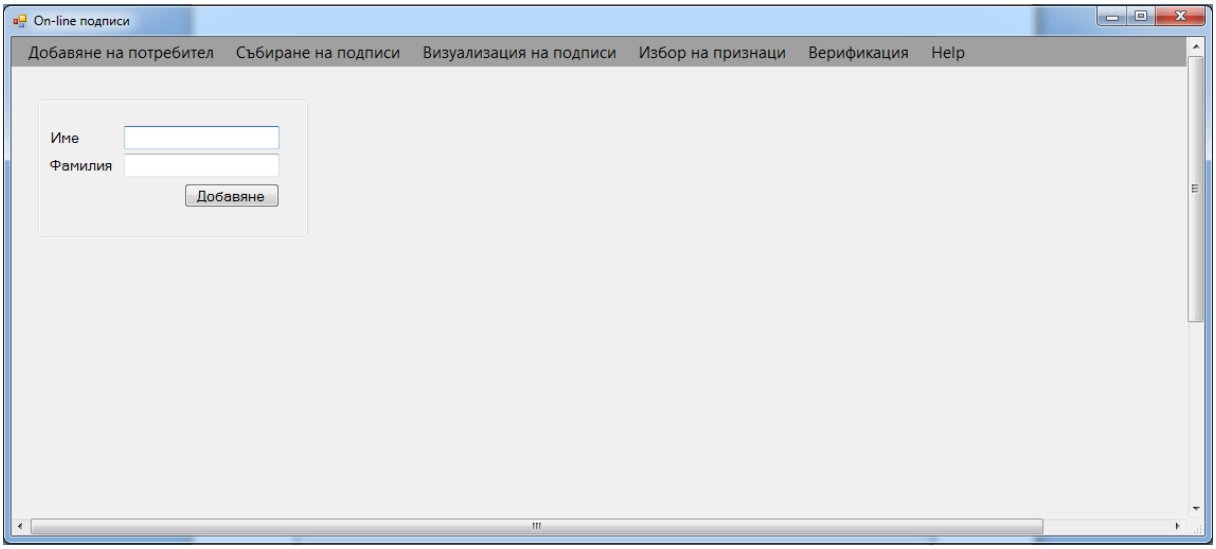
Разработено е софтуерно приложение със следната функционалност: добавяне на потребител, търсене на определен потребител чрез навигация, събиране на подписи от даден потребител, предварителна обработка на данните за подписи, извличане на признаци, избор на признаци, визуализация на собствените и фалшиви подписи на избран потребител, обучение на класификатор и разпознаване по подпис. Следната фигура представя началния екран на приложението.



Фигура 5.3 Начален екран на приложението

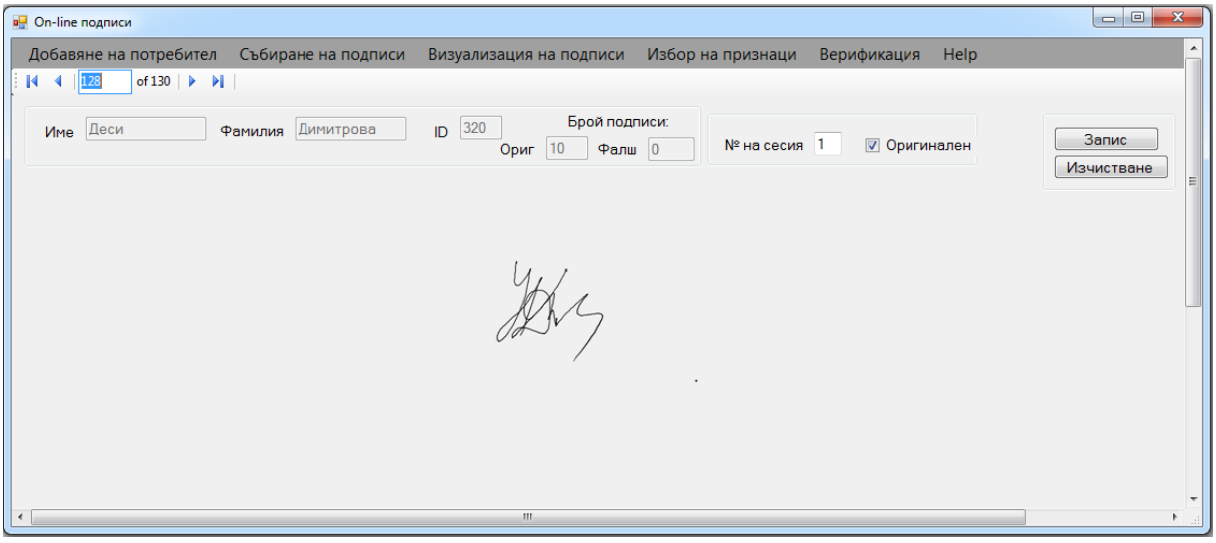
От основното меню включва следните функционалности:

1. *Добавяне на потребител* – след въвеждането на име и фамилия на потребител, записът се добавя в базата данни в таблицата *User*;



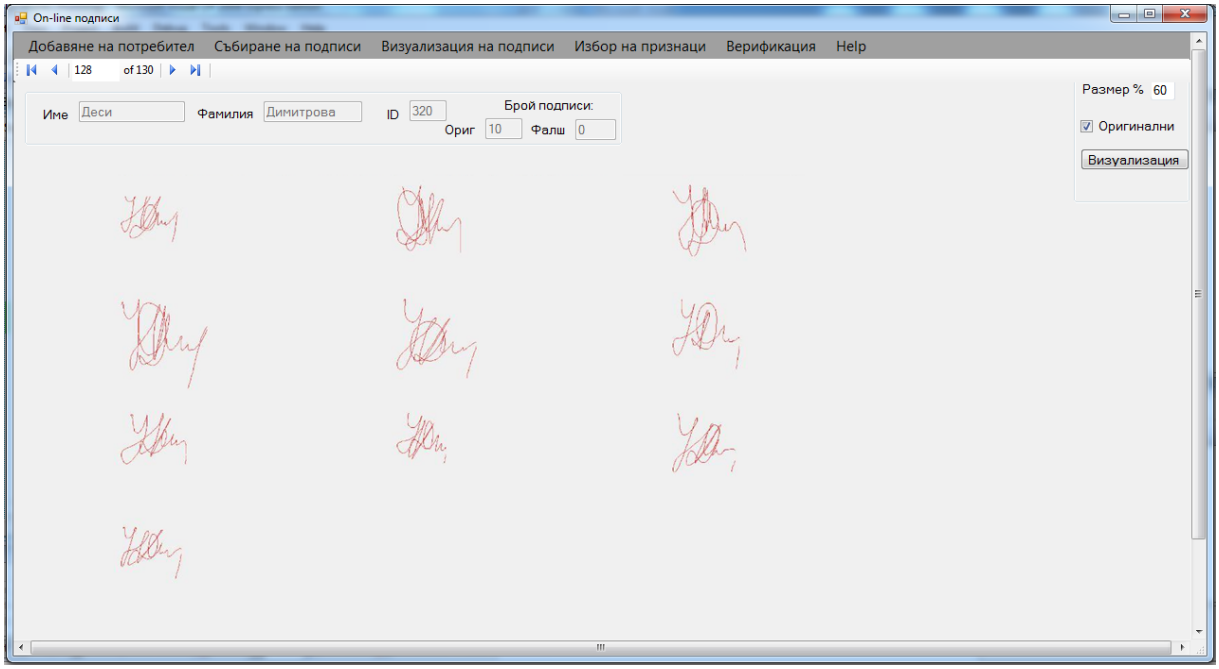
Фигура 5.4 Екран за добавяне на потребител

2. *Събиране на подписи* – налице е възможност за търсене на даден потребител чрез навигация напред и назад по записите в таблицата *User*. За избран потребител се показва броя на събраните негови собствени и умели подправени подписи. Придобавяне на подписи, операторът посочва дали полаганият подпис е собствен или не.



Фигура 5.5 Екран за събиране на подписи

3. *Визуализация на събраните подписи* на избран потребител – В този екран може да се посочи дали да се визуализират собствените или подправените подписи на избран потребител, както и размера им, изразен в процент от този на съхранените подписи в базата данни за подписи.



Фигура 5.6 Екран за визуализиране на подписи

4. *Избор на признаци и обучение* –За избран потребител се извършват последователно следните стъпки:

   а) *Извличане на признаци* - в директорията, посочена в полето *Път до файла* се създават три текстови файла (по един за собствените подписи, умелите и неумелите фалшификати). В редовете им се съдържат стойностите на признаците за всеки от съответния вид подписи. Същевременно се записва още един текстов файл, съдържащ групи признаци с висока взаимна корелация и нейната стойност;

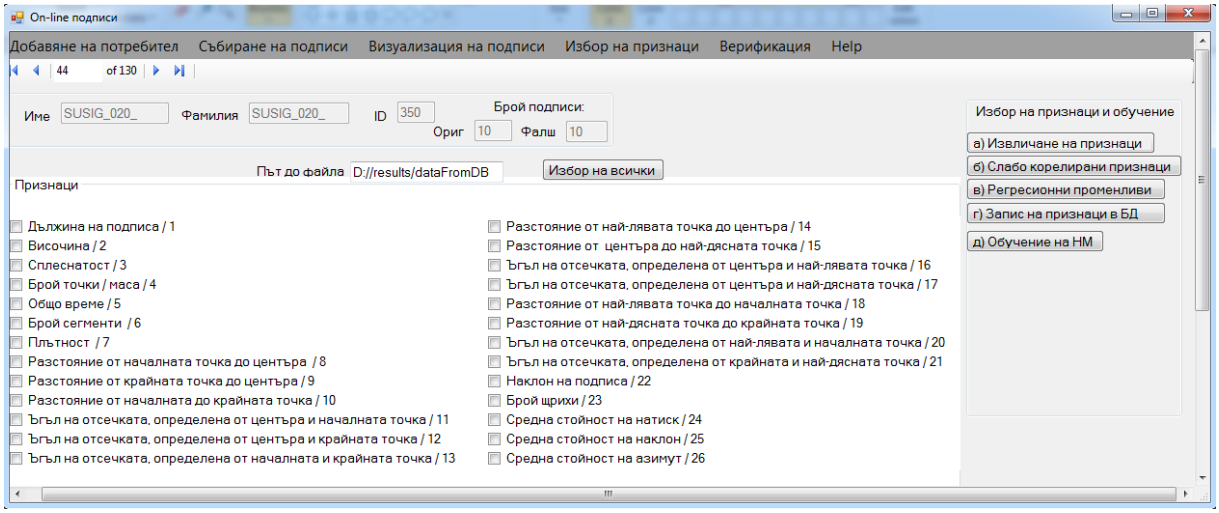
   б) *Слабо корелирани признаци* – операторът натиска този бутон след като е запазил един от всяка група признаци (като е премахнал останалите) и е поставил отметка пред него, в резултат на което стойностите на редуцираните признаци се записват в нови три текстови файла;

   в) *Регресионни променливи* – извършва се избор на регресионни променливи, като резултатите (стойност на *Cp*) от прилагането на метода за различните стойности на p ($p = k - r$) и r ($3 < r < 13$) се записват в отделен файл. От оператора се очаква да вземе решение относно това кои признаци да останат въз основа на

резултата от прилагането на метода за избор на признаци и да постави отметки пред тях;

г) *Запис на признаци в БД* - Избраните признаци се записват в базата данни в таблицата *UserFeature*;

д) *Обучение на НМ* – провежда се обучение на 30 на брой невронни мрежи. Избира се един от моделите (Таблица 4.9), който демонстрира най-висока точност и съответния брой скрити неврони чрез метода на 10-кратна крос валидация. Съответният модел се обучава и запаметява за участника.



Фигура 5.7 Екран избор на признаци

5. *Верификация на самоличност* – Участникът избира от падащото меню имената си и полага подписа си. За подписа, който се тества, се извличат съответните признаци (според таблицата *UserFeature*) и се подават на запаметения класификатор, който дава отговор дали самоличността се потвърждава или отхвърля.
6. *Help* – оттук се предоставя възможност на оператора по всяко време да отвори наръчника за работа с основното меню на системата във формат на текстов файл.

5.4 Резултати в Глава 5

1. Разработен е протокол за събиране и обработка на он-лайн подписи с графичен таблет.
2. Разработена е база данни за он-лайн подписи.
3. Разработен е прототип на цялостна софтуерна система за верификация на он-лайн подписи.

Заключение

В настоящата дисертационна работа е предложен комбиниран метод за разпознаване на он-лайн подписи, който е реализиран в софтуерно приложение. Разгледан е случая на верификация на подпис, поради все по-широкото му използване в обществената практика. Обхванат е целия процес по създаването на система за разпознаване по подписи с всички необходими за целта стъпки, включващи най-общо събиране на подписи, избор на признаци и създаване на модели за верификация. Изследвано е значението на типа подправени (умели и неумели) подписи както за избора на признаци, така и за самата верификация. Експериментите са проведени върху създадена за целта база от данни за подписи от осем участника, както и върху базата от данни за подписи SUsig, съдържаща подписи от 89 участника. Постигнатите резултати по време на експериментите демонстрират задоволителна точност на разпознаване от 98,46 %. Извършено е сравнение на получените резултати с класификатор НМ и класификатор по *k*-най-близки съседи. По такъв начин поставените в дисертационната работа цели задачи са изпълнени напълно.

Разработките и изследванията в областта на разпознаването на он-лайн подписи ще продължат и в бъдеще и ще бъдат насочени: 1) към усъвършенстване и подобряване на работата на системата като степен на автоматизация и точност; 2) тестването на системата върху различни бази данни и с използването на други признаци; 3) поради все по-широкото навлизане на такива системи в практиката и възможността за измами все по-често ще възниква въпросът за установяването на фалшификати. В тази връзка от голяма полза ще бъде добавянето на телевизионна камера, която ще отчита допълнително характеристиките и движението на ръката на подписващия се. Очертава се възможност и за съвместна изследователска работа с експерти от НИКК при МВР и участие в национални и международни проекти.

Приноси в дисертацията

Научно-приложни приноси:

- Предложен е подход за минимизиране на признаково пространство чрез последователно прилагане на метода на корелационните плеяди и регресионния анализ. С прилагането му е получено минимално подмножество от признаци за всеки потребител, включен в база данни;
- Предложен е метод за създаване на класификатор като комбинация от индивидуални за всеки потребител класификатори;
- Предложен е обобщен мрежов модел на процеса за он-лайн верификация на подписи;
- Извършено е сравнение на точността на верификацията с използването на умели и неумели фалшификати в процеса на обучението;
- Извършено е сравнение на точността на верификацията по общи и индивидуални признаци.

Приложни приноси:

- Разработена е база данни и протокол за събиране на он-лайн подписи;
- Разработен е прототип на софтуерна система, даваща възможност за: добавяне на потребител, търсене на даден потребител, събиране на подписи с графичен таблет, предварителна обработка на данните за подписи, извличане на признаци, избор на признаци, визуализация на собствени и фалшиви подписи, избор на модел на класификатор и обучението му;
- Създаден е речник на част от специфичните термини в разпознаването на образи на български и английски език.

Публикации по темата

1. Boyadzhieva D., Gluhchev G., *Feature Set Selection for On-line Signatures using Selection of Regression Variables*, In: Proceedings of the 4th International Conference on Pattern Recognition and Machine Intelligence PReMI'11, June 27-July 01 2011, Moscow, Russia, LNCS6744, Springer-Verlag Berlin Heidelberg, 2011, ISSN 0302-9743, pp. 440-445, http://link.springer.com/chapter/10.1007/978-3-642-21786-9_71#page-1.
2. Boyadzhieva D., Gluhchev G., *An approach to feature selection for on-line signatures*, In: Proceedings of the 10th Anniversary International Scientific Conference UNITECH'10 Vol. I, Nov. 19-20, 2010, Gabrovo, Bulgaria, ISSN 1313-230X, pp. I-457…I-460.
3. Бояджиева Д., *Извличане на признаци на подпис от графичен таблет*, В: Научни трудове на Русенски университет "Ангел Кънчев", 2010, том 49, серия 3.2, Русе, България, ISBN 1311-3321, стр. 101-105, http://conf.uni-ruse.bg/bg/docs/cp10/3.2/3.2-18.pdf.
4. Dimitrova D., Gluhchev G., *Pressure Evaluation in On-Line and Off-Line Signatures*, In: Proceedings of the Joint COST 2101 & 2102 International Conference on Biometric ID Management and Multimodal Communication (BioID_MultiComm'09), Sep. 16-18 2009, Madrid, Spain, LNCS 5707, Springer -Verlag Berlin Heidelberg, 2009, ISSN 0302-9743, pp. 207-211, http://link.springer.com/chapter/10.1007/978-3-642-04391-8_27#page-1.
5. Desislava Boyadzhieva and Georgi Gluhchev, *A GN model for on-line signature verification*, In: Proceedings of the 14th Int. Workshop on Generalized Nets, Burgas, Bulgaria, 29-30 November 2013, ISSN 1313-6860, pp. 65-70, http://www.ifigenia.org/w/images/f/f6/IWGN2013-14-65-70.pdf.
6. Boyadzhieva D., Gluhchev G., *A Combined Method for On-Line Signature Verification* (приета за печат в списанието *Cybernetics and Information Technologies*, кн. 2, 2014 г.).

Списък на други публикации

1. Dimitrova D., Popov A., Finding face features in color images using fuzzy hit-or-miss transform, In: Proceedings of the 9th WSEAS International Conference on Fuzzy Systems (FS'08), Technical University of Sofia, 2008, pp. 79-84, ISBN 978-960-6766-57-2, http://www.wseas.us/e-library/conferences/2008/sofia/FS/fs-11.pdf, .
2. Popov A., Dimitrova D., A new approach for finding face features in color images, In: Proceedings of the 4th International IEEE Conference on Intelligent Systems (IS'08) Vol. 2, Varna, 2008, pp.12-33 - 12-37, ISBN 978-1-4244-1739-1.

Библиографска справка

[1] Plamondon, R., Lorette, G.: *Automatic signature verification and writer identification – the state of the art*. Pattern Recognition 22, pp. 107–131, 1989.

[2] Plamondon, R., Srihari, S. N.. *On-line and off-line handwriting recognition: A comprehensive survey*. IEEE Trans. PAMI, 22(1):pp. 63–84, 2000.

[3] J. Fierrez-Aguilar, L. Nanni, J. Lopez-Penalba, J. Ortega-Garcia, and D. Maltoni. *An online signature verification system based on fusion of local and global information*. In Proc. of IAPR Intl. Conf. on Audio- and Video-Based Biometric Person Authentication, AVBPA, pp. 523–532. Springer LNCS-3546, 2005.

[4] Gupta, G. K., "*The State of the Art in On-line Handwritten Signature Verification*", Monash University, Australia, 2006.

[5] Yeung D. et al., *SVC2004: First International Signature Verification Competition*, Proceedings of the International Conference on Biometric Authentication (ICBA), Hong Kong, 2004, pp.16-22.

[6] Kholmatov A., YanikogluB.. *Identity authentication using improved online signature verification method*. Pattern Recognition Letters, 26(15):2400–2408, 2005.

[7] L. L. Lee, T. Berger, and E. Aviczer. *Reliable on-line human signature verification systems*. IEEE Trans. on PAMI, 18(6):643–647, 1996.

[8] J. Richiardi et al. *Local and global feature selection for on-line signature verification*. In Proc. ICDAR, Seoul, Korea, August-September 2005.

[9] J. Fierrez-Aguilar, N. Alonso-Hermira, G. Moreno-Marquez, and J. Ortega- Garcia. *An off-line signature verification system based on fusion of local and global information*. In Proc. BIOAW, pp. 295–306. Springer LNCS-3087, 2004.

[10] Chatterjee, S., Hadi, *A.: Regression Analysis by Example*. 4th Ed. New York, 2006.

[11] D. Impedovo et al. *Automatic signature verification – The state of the art*, IEEE Transaction on Systems, Man and Cybernetics part C, pp. 609-635, Vol. 38, Issue 5, 2008.

[12] J. Richiardi, H. Ketabdar, A. Drygajlo, *Local and Global Feature Selection for On- line Signature Verification*, IEEE 8th International Conference on Document Analysis and Recognition (ICDAR'05), vol.2, pp. 625 – 629, 2005.

[13] P. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. New York: Wiley, 1973.

[14] Stan Z. Li, Anil K. Jain (Eds.): *Encyclopedia of Biometri*cs. Springer, 2009.

[15] Plamondon, R. ed.: *Progress in automatic signature verification*. World Scientific, Singapore, 1994.

[16] Fierrez-Aguilar, J., Ortega-Garcia, J.: *On-line signature verification*. In Jain, A., Flynn, P., Ross, A. (eds.): Handbook of Biometrics, pp. 189–209. Springer, New York, 2008.

[17] Nalwa, V.: *Automatic on-line signature verification.* Proc. IEEE 85, pp.215–239,1997

[18] Jain, A., Griess, F., Connell, S.: *On-line signature verification*. Pattern Recognit. 35, pp. 2963–2972, 2002

[19] Георги Глухчев, *Класификация тканей по клеточным мазкам (математические алгоритмы)*: диссертация канд. физ.-мат. наук / [МГУ], Фак. вычислит. математики и кибернетики, Москва 1977.

[20] H. Baltzakis and N. Papamarkos, "*A new signature verification technique based on a two-stage neural network classifier,*" Engineering Applications of Artificial Intelligence, vol. 14, no. 1, pp. 95-103,2001.

[21] L. Xu, A. Krzyzak, C.Y. Suen, *Methods of combining multiple classifiers and their applications to handwriting recognition*, IEEE Transactions on Systems, Man, and Cybernetics 22 (3), pp. 418–435, 1992.

[22] J. Kittler, M. Hatef, R. P. Duin, and J. G. Matas, "*On combining classifiers,*" IEEE Transactions on Pat. Anal. and Mach. Intel., vol. 20, pp. 226–239, 1998.

[23] C. Sansone, M. Ven, *Signature Verification: Increasing Performance by a Multi-Stage System*, Pattern Analysis & Applications, pp: 169–181, 2000.

[24] L. Likforman-Sulem, S. Garcia-Salicetti, J. Dittmann, J. Ortega-Garcia, N. Pavesic, G. Gluhchev, S. Ribaric, and B. Sankur, *Report on the hand and other modalities stated of the art,* Biometrics for Secure Authentication, 2005 .

[25] Desislava Dimitrova, Georgi Gluhchev. *Pressure Evaluation in On-Line and Off-LineSignatures*. COST 2101/2102 Conference, pp.207-211, 2009.

[26] Julian Fierrez, Javier Ortega-Garcia, Daniel Ramos, Joaquin Gonzalez-Rodriguez. *HMM-based on-line signature verification: Feature extraction and signature modeling*. Pattern Recognition Letters 28:16, 2325-2334, 2007.

[27] Berrin A. Yanikoglu, Alisher Kholmatov: *Online Signature Verification Using Fourier Descriptors*. EURASIP J. Adv. Sig. Proc., 2009.

[28] J. Fierrez Aguilar, S. Krawczyk, J. Ortega Garcia, A. K. Jain, *Fusion of local and regional approaches for on-line signature verification* In Proc. of Intl. Workshop on Biometric Recognition Systems, pp. 188-196, 2005.

[29] LaMotte, L.R., Hocking, R.R.: *Computational Efficiency in the Selection of Regression Variables*. Technometrics. 12, pp. 83-93, 1970.

[30] Nelson, W., Turin, W. and Hastie, T., "*Statistical Methods for On-line Signature Verification*" International Journal of Pattern Recognition and Artificial Intelligence, Vol. 8, No. 3, pp. 749-770, 1994.

[31] Musa Mailah and Lim Boon Han. *Biometrics signature verification using pen position, time, velocity and pressure parameters*. Jurnal Teknologi,UTM 48(A), pp. 35 – 54, 2008.

[32] Nyssen E, Sahli H, Zhang K. *A multi-stage online signature verification system.* Pattern Analysis and Applications, 5(3): pp. 288-295, 2002.

[33] X. Yang, T. Furuhashi, K. Obata, and Y. Uchikawa, "*Constructing a high performance signature verification system using a GA method*," in ANNES 95, 1995.

[34]]C.M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, 1997

[35] B. Fuentes, M. Garcia-Salicetti, S. Dorizzi, "*Online signature verification: fusion of a hidden markov model and a neural network via a support vector machine*", 8th International Workshop on Frontiers in Handwriting Recognition, FHR02, pp. 253-258, 2002.

[36] Fierrez-Aguilar, J., Ortega-Garcia, J., Gonzalez-Rodriguez, J. *Target dependent score normalization techniques and their application to signature verification*. IEEE Trans. on Systems, Man, and Cybernetics – Part C 35 (3), pp. 418–425, 2005.

[37] Liang Wan, Bin Wan, Zhou-Chen Lin "*On-Line Signature Verification with Two-Stage Statistical Models*", Eighth International Conference on Document Analysis and Recognition (ICDAR'05), pp. 282 – 286, 2005.

[38] Boyadzhieva D., Gluhchev G., *Feature Set Selection for On-line Signatures using Selection of Regression Variables*, In: Proceedings of the 4th International Conference on Pattern Recognition and Machine Intelligence PReMI'11, pp. 440-445, 2011.

[39] Ortega-Garcia, J., Fierrez-Aguilar, J., Martin-Rello, J., Gonzalez-Rodriguez, J.*: Complete signal modeling and score normalization for function-based dynamic signature verification*. In: Proc. of IAPR Intl. Conf. on Audio- and Video-based Person Authentication, AVBPA, Springer, pp. 658–667, 2003.

[40] Lim, Boon Han and Mailah, Musa. *Intelligent biometric signature verification system incorporating neural network*.Jurnal Mekanikal (20). pp. 22-41, 2005.

[41] Reena Bajaj, Santanu Chaudhury, *Signature verification using multiple neural classifiers*, Pattern Recognition, Vol. 30, No. 1, pp. 1-7, 1997.

[42] George Azzopardi and Kenneth P Camilleri, "*Off-line Handwritten Signature Verification using Radial Basis Function Neural Networks*," IEEE International Conference on Electrical and Electronics (INDICON-2008), pp. 17-22, 2008.

[43] Z.-H. Quan and K.-H. Liu. *Online signature verification based on the hybrid HMM/ANN model*. Int. J. Comput. Sci. Netw. Secur., 7(3):313–321, 2007.

[44] Szklarzewski, A. , Derlatka, M., *On-line Signature Biometric System with Employment of Single-output Multilayer Perceptron*, Biocybernetics and Biomedical Engineering, Volume 26, Number 4, pp. 91–102, 2006.

[45] J. Ortega-Garcia, J. Fierrez-Aguilar, J. Martin-Rello, and J. Gonzalez-Rodriguez. *Completesignal modeling and score normalization for function-based dynamic signature*

verification. In Proc. International Conference on Audio- and Video-Based Biometric Person Authentication 2003, pp. 658–667, 2003.

[46] L. Kuncheva, *Combining pattern classifiers, Methods and Algorithms*, Wiley, New York, 2004.

[47] M. Gori and F. Scarselli, *Are multilayer perceptrons adequate for pattern recognition and verioccation?*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, no. 11, pp. 1121-1132, 1998.

[48] Ho, T.K.: *Multiple classifier combination: Lessons and next steps*. In Bunke, H., Kandel, A., eds.: Hybrid Methods in Pattern Recognition. World Scientific, 2002.

[49] Neural Network Toolbox™ 7, http://www.mathworks.com/help/pdf_doc/nnet/nnet_ug.pdf

[50] G. P. Zhang, "*Neural networks for classification: a survey*," Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, vol. 30, pp. 451-462, 2000.

[51] McCabe, Alan, Trevathan, Jarrod, and Read, Wayne. *Neural network-based handwritten signature verification*. Journal of Computers, 3 (8). pp. 9-22, 2008.

[52] Xuhong Xiao, Graham Leedham, *Signature verification by neural networks with selective attention*. Journal Applied Intelligence, Vol.11, No.2, pp 213-223, 1999.

[53] M. Freire-Santos, J. Fierrez-Aguilar, and J. Ortega-Garcia. *Cryptographic key generation using handwritten signature*. In Proc. of SPIE, volume 6202, pp. 225–231, 2006.

[54] Hocking, R.R.: *The Analysis and Selection of Variables in Linear Regression*. Biometrics. 32,1976.

[55] Hocking, R.R., Leslie, R.n.*: Selection of the Best Subset in Regression Analysis*., Technometrics. 9, 531-540,1967.

[56] Douglas C. Montgomery, Elizabeth A. Peck, G. Geoffrey Vining, *Regression analysis by example*, 4th ed, Wiley, 2007.

[57] *Mallows*, C.L. *Some Comments on CP*. Technometrics 15 (4): 661–675, 1973.

[58] http://www.microsoft.com/visualstudio/en-us/products/2010-editions/express

[59] http://www.microsoft.com/sqlserver/en/us/editions/compact.aspx

[60] http://www.microsoft.com/download/en/details.aspx?displaylang=en&id=20039

[61] http://www.wacom-asia.com/intuos3/intuos3_630.html

[62] http://www.mathworks.com/products/matlab/

[63] http://www.mathworks.com/matlabcentral/answers/22944-neural-network-design

[64] Neural network toolbox documentation http://www.mathworks.com/help/nnet/index.html

[65] Alisher Kholmatov, Berrin A. Yanikoglu: *SUSIG : an on-line signature database, associated protocols and benchmark results*. Pattern Anal. Appl. 12(3): 227-236, http://biometrics.sabanciuniv.edu/SUSIG, 2009 .

[66] LeCun, L. Bottou, G. Orr and K. Muller: *Efficient BackProp*, in Orr, G. and Muller K.

(Eds), Neural Networks: Tricks of the trade, Springer, 1998.

[67] Dougherty G., *Pattern Recognition and Classification*, Springer-Verlag New York Inc. ISBN: 9781461453222.

[68]http://www.mathworks.com/matlabcentral/answers/25759-normalizing-data-for-neural networks.

[69] http://www.mathworks.se/matlabcentral/answers/57702#answer_83202.

[70] Нишева, М, Д. Шишков. *Изкуствен интелект*. "Интеграл" - Добрич, 1995.

[71] Kurt Hornik *Approximation Capabilities of Multilayer Feedforward Networks*, Neural Networks, 4(2), pp. 251–257, 1991.

[72] A. J. C. Sharkey, *Multi-net systems*, in Combining Artificial Neural Nets: Ensemble and Modular Multi-Net Systems, A. J. C. Sharkey, Ed. Berlin, Springer-Verlag, pp. 1–30, 1999.

[73] M. P. Perrone and L. N. Cooper, *When networks disagree: Ensemble methods for hybrid neural networks*, in Neural Networks for Speech and Image Processing, R. J. Mammone, Ed. London, U.K.: Chapman & Hall, pp. 126–142, 1993.

[74] Arun Ross and Anil K. Jain. *Multimodal biometrics: An overview.* In Proc. 12th EuropeanSignal Processing Conf., pages 1221–1224, 2004.

[75] Fumera, G., Roli, F. *A theoretical and experimental analysis of linear combiners for multiple classifier systems*. IEEE Trans. Pattern Anal. Machine Intell. 27 (6), pp. 942–956, 2005.

[76] S.B. Cho and J.H. Kim, *Combining Multiple Neural Networks by Fuzzy Integral for Robust Classification*, IEEE Trans. Systems, Man, and Cybernetics, vol. 25, no. 2, pp. 380-384, 1995.

[77] A. K. Jain, S. Prabhakar and A. Ross, *Fingerprint Matching: Data Acquisition and Performance Evaluation*, MSU Technical Report TR99-14, 1999.

[78]Turner, K., Ghosh, J.*: Error correlation and error reduction in ensemble classifiers*. Connection Science. 8 nos. 3 & 4, pp. 385-404,1996.

[79] Maltoni, D. Maio, A. K. Jain, and S. Prabhakar. *Handbook of Fingerprint Recognition*. Springer, 2003.

[80] K. Hornik, M. Stinchcombe, and H. White, *Multilayer feedforward networks are universal approximators*, Neural Networks, vol. 2, pp.359–366, 1989.

[81] Tebelskis, J., *Speech Recognition Using Neural Networks*, PhD Dissertation, Carnegie Mellon University, 1995.

[82] "*Neural Network-Based Face Detection*", Henry A. Rowley, Shumeet Baluja, Takeo Kanade, IEEE Trans. Pattern Anal. Mach. Intell., vol. 20 (1998), pp. 23-38.

[83] *Approximation by Superpositions of a Sigmoidal Functionby*: G. Cybenko, In: Mathematics of Control, Signals, and Systems, Vol. 2, pp. 303—314, 1989.

[84] Park, J., Wsandberg, J. *Universal approximation using radial basis functions network.* Neural

Comput. 3 (2), pp. 246–257, 1991.

[85] Lew, J.S.,*An Improved Regional Correlation Algorithm for Signature Verification which Permits Small Speed Changes Between Handwriting Segments*, IBMRD(27), No. 2, pp. 181-185, 1983.

[86] Stuart Russell, Peter Norvig, "*Artificial Intelligence: A Modern Approach*, 3rd Edition" Prentice Hill, 2009.

[87] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery, Numerical Recipes: *The Art of Scientific Computing,* Third Edition, 1235 pp. + xxi, 2007.

[88] T. G. Dietterich. *Approximate statistical tests for comparing supervised classification learning algorithms*. Neural Computation, 7(10):1895–1924, 1998.

[89] A. Sharkey, N. Sharkey, *How to improve the reliability of artificial neural networks*, Technical Report CS-95-11, Department of Computer Science, University of Shefield,1995.

[90] Lars K. Hansen, Peter Salamon, *Neural Network Ensembles*, IEEE Transactions on PAMI, Vol. 12, No. 10, pp. 93-1001, 1990.

[91] Kuncheva L.I. *Combining classifiers: Soft computing solutions*, in: S.K. Pal and A. Pal (Eds.) Pattern Recognition: From Classical to Modern Approaches, World Scientific Publishing Co, pp. 427-452, 2001.

[92] T. K. Ho, "*The random subspace method for constructing decision trees*," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, no. 8, pp. 832–844, 1998.

[93] L.I. Kuncheva, F. Roli, G.L. Marcialis and C.A. Shipp, "*Complexity of Data Subsets Generated by the Random Subspace Method: an Experimental Investigation*", MCS 2002.

[94] M.D.Richard, R.P.Lippmann, "*Neural network classifiers estimate Bayesian a posteriori probabilities*", 1991.

[95] W. Wei, T. K. Leen, and E. Barnard. *A fast histogram-based postprocessor that improvesposterior probability estimates*. Neural Computation, 11(5):1235–1248, 1999.

[96] R. Lippmann, *Pattern classification using neural networks*, IEEE Commun. Magazine, vol. 27, pp. 47–64, 1989.

[97] Ink Data: http://msdn.microsoft.com/en-us/library/ms811395.aspx

[98] Critical Values for Pearson's Correlation Coefficient: http://capone.mtsu.edu/dkfuller/tables/correlationtable.pdf

[99] С.А. Айвазян, З.И. Бежаева, О.В. Староверов, *Классификация многомерных наблюдений*, Москва, Статистика, 240 стр., 1974.

[100] Fawcett, T., 2003. *ROC graphs: Notes and practical considerations for researchers, Tech Report HPL-2003-4*, HP Laboratories.http://www.hpl.hp.com/techreports/2003/HPL-2003-4.pdf.

[101] Gluhchev G., M. Savov, O. Boumbarov, D. Vassileva. *A New Approach to Signature Based*

Authentication, 2nd Int. Conf. on Biometrics, pp. 594-603, 2007.

[102] Savov M., G. Gluhchev. *Signature verification via Hand-Pen motion investigation*, Proc. Int. Conf. Recent Advances in Soft Computing, pp. 490-495, 2006.

[103] Atanassov, K., G. Gluhchev, S. Hadjitodorov, A. Shannon, V. Vassilev, *Generalized Nets and Pattern Recognition*. KvB Visual Concepts Pty Ltd, Monograph No. 6, Sydney, 2003.

[104] *A generalized net model of the process of handwriting identification.* In: [103].

[105] Atanassov K., *Generalized Nets*, World Scientific, Singapore, New Jersey, London, 1991.

[106] Schmidt T., Riffo V., Mery D.: *Dynamic Signature Recognition Based on Fisher Discriminant*. CIARP, volume 7042 of Lecture Notes in Computer Science, Springer, pp.433-442, 2011.

[107] Abdulla Ali, *Offline Signature Verification using Radon Transform and SVM/kNN Classifiers*, ISSN 0136-5835, Вестник ТГТУ, Том 15. № 1, 2009.

[108] Meenakshi K., Sargur S., Aihua X., *Offline Signature Verification And Identification Using Distance Statistics*, In: International Journal of Pattern Recognition and Artifficial Intelligence, Vol. 18, No. 7, pp.1339-1360, 2004.