



M Ű E G Y E T E M 1 7 8 2

Budapesti Műszaki és Gazdaságtudományi Egyetem
Villamosmérnöki és Informatikai Kar
Automatizálási és Alkalmazott Informatikai Tanszék

Szücs Cintia Lia

**ON-LINE ALÁÍRÁS-
HITELESÍTÉSI ALGORITMUSOK
VIZSGÁLATA**

KONZULENS

Dr. Kővári Bence

BUDAPEST, 2018

Tartalomjegyzék

Összefoglaló	4
Abstract.....	5
1 Bevezetés	6
1.1 Aláírás-hitelesítés feladata	6
1.2 Aláírás-hitelesítés jelentősége.....	7
1.3 Aláírás-hitelesítés típusai	8
1.3.1 On-line aláírás-hitelesítés	8
1.3.2 Off-line aláírás-hitelesítés.....	8
2 Irodalmi áttekintés.....	10
2.1 Aláírások vizsgált tulajdonságai	10
2.2 Előfeldolgozás	11
2.3 Hasonlóság mérése	12
2.3.1 Algoritmusok	12
2.3.2 Távolságmértékek.....	14
2.4 Eredmények értékelése	14
2.4.1 False Rejection Rate (FRR)	15
2.4.2 False Acceptance Rate (FAR).....	15
2.4.3 Equal Error Rate (EER)	15
2.4.4 Average Error Rate (AER)	16
2.5 Adatbázisok	16
2.5.1 Signature Verification Competition 2004 (SVC2004)	17
3 Használt megközelítésem	19
3.1 Alapmegközelítés.....	19
3.1.1 DTW algoritmus megvalósítása.....	19
3.1.2 Vizsgált aláírás-tulajdonság kiválasztásának háttere	20
3.2 Osztályozók típusai.....	22
3.3 Minimális AER érték elérését célzó osztályozás	23
3.3.1 Tanulási fázis	23
3.3.2 Tesztfázis	24
3.4 Automatikus osztályozás	24
3.4.1 Tanulási fázis	24

3.4.2 Tesztfázis	25
3.5 DTW és Warping Path Score kombinációja	25
3.5.1 Warping Path Score lényege	25
3.6 Alírási időbeli hosszának felhasználása	28
3.6.1 Súlyozott, összetett osztályozó	31
3.7 Alírási tulajdonságok	32
3.7.1 Származtatott tulajdonságok	33
3.7.2 Tulajdonságok kombinációi	35
3.8 Előfeldolgozás használata	38
4 Tesztelést segítő alkalmazás	42
4.1 Implementált algoritmusok eredményeinek vizsgálata	42
4.2 Osztályozások eredményességének vizsgálata	42
4.2.1 Konfigurációs lehetőségek	42
4.2.2 Osztályozás kimenete	43
5 Mérési eredmények	44
5.1 Eredmények értékelése	48
6 Összegzés	49
7 Irodalomjegyzék	50

Összefoglaló

Felgyorsult, digitalizált világunkban, ahol a papíralapú szerződéseket, egyéb dokumentumokat egyre szélesebb körben váltják fel azok elektronikus megfelelői, természetesen merül fel az igény az aláírás-hitelesítés automatizálására. Célunk korlátos időn belül, minél nagyobb biztonsággal programozott döntést hozni egy-egy kézi aláírás eredetiségéről.

Az aláírások két eltérő módon állhatnak rendelkezésünkre, miszerint tárolhatjuk a kézzel írt aláírásoknak a képét vagy egy digitális segédeszköz (táblagép, kamera, speciális toll) használatával rögzített, több dimenziós adathalmaznak is tekinthetjük őket. Eszerint a tématerületet off-line és on-line alterületekre bonthatjuk.

Jelen dolgozat az on-line területtel foglalkozik, azon belül is DTW (Dynamic Time Warping) algoritmus használatán alapuló hitelesítési módszerekkel. A dolgozat célja hitelesítést befolyásoló tényezők vizsgálata, úgymint a felhasznált leíróparaméterek megválasztása, a határérték meghatározásának módja vagy a hasonlóság leírására szolgáló különböző algoritmusok alkalmazása. A vizsgálatok törekvése a befolyásoló tényezőkről és azok összefüggéséről olyan képet adni, mely jó kiindulási alapot jelent aláírás-hitelesítő rendszerek tervezéséhez.

Az egyes tényezők helyes megválasztásával sikerült AER elérhető, elméleti, alsó korlátját az alapmegközelítés által eredményezett 19,125%-ról 2,875%-ra csökkenteni.

Abstract

Nowadays most of the traditional paper-based documents are replaced with their digital equivalents and also the demand is increasing for the automated signature verification which helps to programmatically decide about a signature's genuineness. The decision has to be made within a limited time and with high accuracy. The target is to build an accurate and trusted verification system.

The signatures can be available in two different ways. Either it can be stored as pictures of the handwritten signature or as a multidimensional dataset which is recorded using some of the special digital devices. This partitioning splits the signature verification into two areas off-line and on-line.

This paper deals with on-line signature verification, within which it uses DTW (dynamic time warping) algorithm to compare signatures. This paper's goal is to analyse different factors which can influence the accuracy of the verification. Such a factor can be the attributes of the signature which the decision is based on, the used similarity/distance function or the method used to define a threshold which is used during the evaluation phase. This paper tries to specify different influence factors and reveal their relationships regarding the accuracy of the verification. It can be treated as a basis to plan different verification systems.

1 Bevezetés

A kézi aláírás mind a mai napig széleskörben elfogadott, az egyik alapvető biometrikus hitelesítési eszköz [1], mely számos egyedi jellemzőt hordoz magában, amik egyértelműen tulajdonosához köthetnek, csak rá jellemzőek. Alkalmazása mindennapjaink szerves részét képezi, már nagyon régóta használatos például szerződések, pénzügyi papírok, elismervények vagy egyéb fontos dokumentumok ellenjegyzésére.

Csalók azonban mindig is voltak, akik megpróbálnak visszaélni más nevével, személyével, ezért fontos feladattá vált az ilyen hamis aláírások kiszűrése. Szerencsére ez nem lehetetlen feladat. Az írásszakértők feladata, hogy az említett egyedi jellemzők, védjegyek alapján meg tudják különböztetni a hamis aláírásokat az eredetiektől. A világ fejlődésével azonban megfogalmazódott az igény, hogy a szakértőket igénylő analóg meghatározás helyett, automatikusan, aláírás-hitelesítő szoftverek segítségével hozzassunk döntést egy-egy aláírás eredetiségét illetően.

1.1 Aláírás-hitelesítés feladata

Az aláírás-hitelesítés feladata tehát korlátos időn belül, minél nagyobb biztonsággal programozott döntést hozni egy-egy ismeretlen eredetű aláírásról. Ehhez persze szükséges, hogy rendelkezésre álljanak eredeti referencia-aláírások, melyek tanulmányozásával meghatározhatók az aláíróra jellemző egyedi tulajdonságok. A döntés ezek alapján születik meg, ezeket figyelve az új, ismeretlen aláírásokon. Ez az alapmeggondolás minden hitelesítő mögött.

A konkrét megvalósításokat tekintve természetesen rengeteg szempont játszik szerepet a hitelesítőrendszer eredményességében. Számtalan különböző megközelítés létezik, például az aláírás vizsgált tulajdonságait, a hasonlóság meghatározására szolgáló algoritmusokat vagy esetleges előfeldolgozási lépéseket tekintve. Ezek megválasztása nagy hatással lehet az eredményekre. Így sikerült például az egyszerű kiinduló módszer által elért 19,125%-os átlagos hibaarányt a fentiek jó kombinációjának alkalmazásával 2,875%-ra csökkentenem.



1. ábra: Aláírás-hitelesítés nagyon leegyszerűsített lépései

1.2 Aláírás-hitelesítés jelentősége

Az utóbbi időben nagyon fontos kutatási terület, sokan foglalkoznak a témával, számtalan létező megközelítés, próbálkozás van az eredmények javítására. Mi sem bizonyítja jobban a terület jelentőségét, mintsem, hogy vannak különböző versenyek is, melyek évről évre, többször is megrendezésre kerültek. Ezeken különböző aláírás-hitelesítési módszereket használó rendszerek kerültek bemutatásra, végeredményben pedig a győztes kihirdetése mellett publikációk jelentek meg a nevezett rendszerek eredményeiről, működési elveiről. Ezáltal elősegítik a tudomány aktuális állásának nyomon követését.

Az ilyen megmérettetések során nyilvános aláírás-adatbázisok kerültek kiadásra, melyeken tesztelték a rendszereket, ezzel biztosítva az eredmények összemérhetőségét. Néhány ilyen adatbázis egészen hosszúéletű lett, az eredeti célján túl is számtalanszor használták fel, így rengeteg módszer eredményei vethetők össze egymásával. Ez egyfajta egységességet biztosít az aláírás-hitelesítési kutatások értékelésében. Két ilyen versengés releváns a dolgozatom szempontjából, az egyik az SVC2004: First International Signature Verification Competition, melynek az adatbázisát én is használtam a munkám során. Valamint a több évben is megrendezett Competition on Signature Verification and Writer Identification for On- and Off-line Skilled Forgeries (SigWiComp), melynek eredményeit az International Conference on Document Analysis and Recognition (ICDAR) elnevezésű konferencián publikálták az aktuális években.

Mára azonban az iparban is kezdenek megjelenni gyakorlati alkalmazások. Bankok, telefonszolgáltatók vagy egyéb szolgáltatások egyre többen próbálnak áttérni a papírmentes működésre. A szerződéseket már csak digitális formában kapják meg az ügyfelek (például e-mailen keresztül), és már azok aláírása is táblagépek segítségével történik. Ez a folyamat az aláírás-hitelesítési megoldások alkalmazását is magával hozta.

Az ilyen rendszerek pontosságáról, elért eredményeiről érthető módon nem rendelkezünk olyan széles körben elérhető információkkal, mint a kutatási megoldások esetén.

1.3 Aláírás-hitelesítés típusai

Az automatizált aláírás-hitelesítésnek két különálló válfaját különböztethetjük meg. Ennek a megkülönböztetésnek alapjául elsősorban a tesztadatok begyűjtésének módja, valamint az így rendelkezésre álló adatok típusa szolgál. Az aláírások felvételére szolgáló különböző eljárások különböző adatstruktúrát is eredményeznek. Ennek megfelelően az már nem meglepő, hogy a feldolgozási folyamat, annak eszközei is eltérőek a két esetben.

1.3.1 On-line aláírás-hitelesítés

Az on-line aláírás-hitelesítés során az egyes aláírásokról rengeteg információ áll rendelkezésünkre. Ehhez szükségünk van valamilyen speciális eszközre, amelynek segítségével ezeket a tulajdonságokat meghatározhatjuk, rögzíthetjük. Ilyen lehet például egy digitális táblagép vagy egy nyomásérzékeny toll, melyek segítségével rögzíthetjük a toll nyomvonalát. Ezeknek az eszközöknek a segítségével az aláírás dinamikáját is meg tudjuk határozni.

Utóbbi olyan plusz információ a számunkra, amit egy aláírás sima, kétdimenziós megjelenítése nem hordoz magában. Így ennek segítségével többet tudunk az aláírásról, mint mikor csak a képét látjuk. Nagy előnye ennek a területnek, hogy a korábban említett dinamikát nagyon nehéz hamisítani, így ennek ismeretében biztosabb állításokat tehetünk az aláírás-hitelesítés során.

Az on-line aláírás-hitelesítés ígéretes jövő előtt áll. Gondoljunk csak bele milyen új lehetőségeket jelent, amennyiben az aláírásunk azonnal digitális formában is elérhető a hitelesítéshez, rengeteg alkalmazásban teret kaphatna az autentikáció során. A jelszavak és pin-kódok használatát felválthatnánk az on-line aláírással, hiszen érezhetően azt nehezebb ellopni, kitalálni vagy hamisítani. [2] A kutatásom során én is on-line aláírás-hitelesítéssel foglalkozom.

1.3.2 Off-line aláírás-hitelesítés

A korábban bemutatott on-line aláírás-hitelesítéssel szemben az off-line módszer esetében nincs szükségünk semmilyen speciális hardvereszközre. Ebben az esetben nem

vizsgáljuk ugyanis, hogy hogyan készült az adott aláírás, csupán a papírra vetett végeredményt. Nincs másra szükségünk, mint valamilyen szkenerre vagy kamerára, amivel az aláírás képét rögzíteni tudjuk.

A feldolgozás során az aláírásokról csak annyi információ áll rendelkezésünkre, amennyit egy-egy kép magában hordoz. Az on-line esethez képest tehát az elemezhető tulajdonságok szűkebb halmazával tudunk csak dolgozni.

A képként beolvasott adatok esetében sokszor szükség van egy előfeldolgozási folyamatra is. A legtöbb hétköznapi esetben, mikor valamit aláírunk, a nyomtatványokon az aláírás elvárt pontos helyét egy vonallal jelzik számunkra. Az ily módon készült aláírások esetében az előfeldolgozás során ezt a vezető vonalat el kell távolítani például, hogy a hitelesítés folyamata során semmilyen problémát, zavart vagy eredménytorzulást ne okozzon. [3]

Az off-line aláírás-hitelesítést célzó kutatások során több különböző módszert, algoritmust is használnak, ám a cél a legtöbb esetben megtalálni az adott aláíróhoz tartozó egyedi jellemvonásokat, melyek minden aláírásában nagyon hasonlóak és szinte minden esetben jelen vannak. Az eredeti referencia-aláírások alapján meghatározott ilyen, ismertetőjegyeket keressük az aktuálisan hitelesítendő aláírásban is.

Nincs könnyű dolgunk, mikor meg szeretnénk állapítani, hogy az adott aláírás valódi-e, a feladat nehézsége az aláírások sokféleségében rejlik. Sokszor nagyobb különbségek figyelhetők meg egy személy saját aláírásai között, mint az eredeti és egy nagyon jól begyakorolt másolat között.

2 Irodalmi áttekintés

A következő néhány alfejezetben szeretném bemutatni a tudomány jelenlegi állását, ezzel nagy vonalakban egy képet adni a terület mélységeiről. A munkám során alkalmazott, releváns részokról kicsit részletesebben is szót ejtek. A különböző megközelítéseket a hitelesítési folyamat mentén csoportosítva fogom felvázolni, egészen onnan, hogy a nyers aláírások beérkeznek a rendszerbe, odáig, hogy a rendszer eredményeit hogyan értékeljük.

2.1 Aláírások vizsgált tulajdonságai

On-line esetben az általában rendelkezésre álló alaptulajdonságok az X- és Y-koordináták, a nyomás és egy időbélyeg. Ezek azok a paraméterek, melyek mentén az aláírásokat megpróbáljuk összehasonlítani.

Ezen felül adatbázistól függően olykor Z tengelyként beleveszik a toll helyzetét is, általában bináris jellemzőként, vagy fent van a toll és nem éri a „papírt” vagy lent van. Ezen felül például a Signature Verification Competition 2004 (SVC2004) adatbázisban rendelkezésre áll a toll helyzetét leíró további két jellemző is. Ilyen az irányszög (azimuth), ami a toll óramutató járásával megegyező irányú elfordulását jelenti a Z-tengely körül. A másik pedig a magasság (altitude), ami a toll és a pozitív Z-tengely által felfelé bezárt szöget határozza meg.

Továbbá lehetőségünk van az ilyen alaptulajdonságokból származtatva újabb tulajdonságokat meghatározni. Ezek a legkülönfélébb ötletek szerint megvalósíthatók, például az alaptulajdonságokból intervallumreprezentánsok használatával. [4] Ennél kézenfekvőbb eset például a deriváltak használata [5], vagy valahányad rendű különbségek, távolság alapú egyéb mértékek, esetleg különböző szögmértékek alkalmazása [6], de ezeken felül is nagyon széles a választék a vizsgált tulajdonságok tekintetében. [7] Ebből is látszik, hogy nincs a tulajdonságoknak egy kőbe vésett halmaza, idővel újabb és újabb levezetési módszerek jelennek meg, minél értékesebb információk kinyerése, s ezáltal az eredmények javítása érdekében.

A nyers, alaptulajdonságok és az azokból származtatottak nem csupán egyenként, hanem azok tetszőleges kombinációjaként is használhatók. Számos lehetőség van a végeredményben használt tulajdonsághalmaz meghatározására, ami az eredményeket

nagyban befolyásolhatja. [5] A rengeteg kombinációs lehetőség miatt lehetetlen minden esetet tesztelni, ezért fontos lehet megvizsgálni, hogy a különböző tulajdonságok egyedi eredményei milyen kapcsolatban állnak, milyen hatással vannak az őket tartalmazó összetételekkel elérhető értékekre. Nagyban segítené a munkát, ha sikerülne valamilyen korrelációt felfedezni, így tudhatnánk, hogy mely potenciális tulajdonságokat lehet érdemes kombinálni, és a rosszul teljesítőkkal nem is kéne foglalkoznunk.

2.2 Előfeldolgozás

Az előfeldolgozást az teszi szükségessé, hogy az aláírások begyűjtése során sok befolyásoló tényező van, melyek apró változása esetén is torzulhatnak az eredményeink. Az aláírások mérete, elnyújtottsága, a kezdőpontja mind-mind olyanok lehetnek, amik közvetlenül egymás után felvett aláírások esetén is változhatnak, nem még akkor, ha éles rendszerekben különböző időben szerzett aláírások összehasonlítása a feladat.

Az előfeldolgozó lépés célja tehát, hogy ezeket az eredménytorzulást előidézhető különbségeket minimalizáljuk, lehetőleg elimináljuk, ezzel csökkentve az eredeti és hamis aláírások osztályon belüli különbségeit.

Sokszor alkalmaznak normalizálást, hogy a különböző aláírásokhoz tartozó egyes értékek összemérhetőek legyenek. Az aláírások helyzetének meghatározása, a kezdőpont vagy a súlypont/középpont egymásba tolása is gyakran alkalmazott eszköz. [8] [9] Gyakorlatban sokszor előfordul, hogy a felvett aláírások méretükben különböznek, ekkor valamilyen skálázásra van szükség. Szintén problémát jelenthet, ha az aláírások egymáshoz képest el vannak forgatva. Az előfeldolgozási fázis során ezt is orvosolni lehet. [10] Alapvető problémát jelenthet, hogy az aláírások rögzítéséhez használt digitális eszközök valamilyen zajt vihetnek az adatokba, érdemes lehet mindenekelőtt ettől is megszabadulni. [11] Fontos megemlíteni, hogy némely lépések semlegesíthetik, elronthatják a mások mögött álló megfontolásokat, így az eredményeket is. Tisztában kell ezért lenni vele, hogy milyen technikát mikor érdemes alkalmazni, és hogy mik közülük az egymást kizáróak.

Ezek az előkészítő lépések nem csak az eredményesség növelésére szolgálnak, ezáltal a hitelesítő rendszerek, mint szoftverek teljesítménye, hatékonysága, gyorsasága is nőhet.

2.3 Hasonlóság mérése

Következő lépésben nincs is más hátra, mint a már megfelelő formára hozott aláírások összehasonlítása. Ehhez definiálni kell egy módszert, algoritmust, mértéket, aminek segítségével számszerűsíteni lehet két aláírás közt a fennálló hasonlóságot. A cél mindezt úgy megtenni, hogy az eredeti aláírások közt kis értékeket kapjunk, míg eredeti és hamis közt ennél érzékelhetően nagyobbakat.

Off-line és on-line esetben értelemszerűen más módszerek használatosak. A következőkben az on-line aláírás-hitelesítés során nagyon népszerű és széleskörben használt DTW (Dynamic Time Warping) algoritmuson alapuló megoldásokra szeretném helyezni a hangsúlyt. [7]

2.3.1 Algoritmusok

A tisztán DTW algoritmust használó megoldások mellett, számtalan továbbfejlesztés is létezik, melyek az eredeti DTW hatékonyságát hivatottak növelni. Ezt lehet úgyis megvalósítani, hogy a DTW által szolgáltatott értékeket felhasználva, azok mentén az eredeti és hamis aláírások közti határvonal meghúzásához alkalmazunk valamilyen egyéb módszert. Ekkor tehát a tanító fázisban, a referencia-aláírások vizsgálata során szeretnénk elsősorban pontosabb eredmények elérését biztosítani. Egy ilyen megközelítés például a support vector machine (SVM) vagy Bayes osztályozó bevonása, melyekkel az átlagos hibaarány (Average Error Rate, AER) 2,745%, illetve 3,56% értékeket vett fel. Az ezek felhasználásával kifejlesztett rendszer meg is nyerte a 2004-ben rendezett First International Signature Verification Competition (SVC2004) mind a két kategóriáját (Task1, Task2). Az általam is használt Task2 adatbázison 2,9%-os AER értéket sikerült elérnie. [12]

Másik megközelítés lehet az eredeti DTW kombinálása valamilyen más módszerrel. Például a következő cikkben vektor kvantálás (vector quantization VQ) módszerével ötvözték. [13] A DTW hiányosságainak, gyengeségeinek kiküszöbölésére a kombinációs megoldások mellett magának az algoritmusnak a módosítására, továbbfejlesztésére is lehetőség van. Így született új technika például az extreme point warping (EPW), mellyel a dinamikus tulajdonságok bevonása nélkül, DTW-vel elért 33%-os EER (Equal Error Rate) értéket EPW-vel 25,4%-ra sikerült csökkenteni. [14] Szintén egyfajta továbbfejlesztése az SM-DTW, vagyis Stability Modulated Dynamic

Time Warping, amivel a MCYT-100 elnevezésű adatbázison 3,09%-os EER-t sikerült elérni profi hamisítványok mellett. [15]

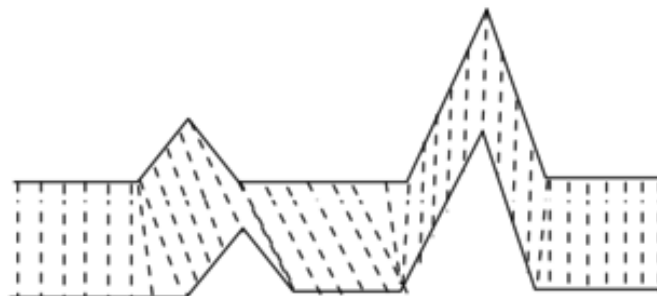
A DTW népszerűsége mellett persze merőben más módszerek is léteznek, mint például Gaussian model description (GAUSSD), Mixture of Gaussians description (MOGD), Nearest-neighbour method description (NND), Principal component analysis description (PCAD), Support vector data description (SVD), Linear programming description (LPD) vagy Parzen window classifier (PWC). Ezek különböző kombinációnak használata is hordozhat magában újabb lehetőségeket. [16]

2.3.1.1 Dynamic Time Warping

Mivel nagyon széles körben elterjedt a DTW algoritmus, és sok aláírás-hitelesítő rendszer alapjául szolgál, ezért ezt szeretném részletesebben is bemutatni. [17]

A DTW algoritmust elsőként beszéd felismerésben alkalmazták, mára azonban számtalan más területen is használatos. Ilyen kutatási területek például az aláírás-hitelesítés – mind az on-line, mind az off-line területen – vagy az adatbányászat és idősorok klaszterezése stb.

A DTW algoritmus feladata, hogy két idősor közötti hasonlóságot meghatározza. Népszerűsége annak köszönhető, hogy mindezt nagyon hatékonyan képes elvégezni, miközben az időbeli elcsúszást és torzulást minimalizálni tudja. Az algoritmus számára adott két értéksorozat, amelyek két idősort reprezentálnak. Az idősorok hossza legyen $N \in \mathbb{N}$ és $M \in \mathbb{N}$, ekkor a DTW $O(N * M)$ idő alatt szolgáltat optimális eredményt. Természetesen ez különböző módszerekkel még fejleszthető, de most csak az algoritmus alapjait tárgyaljuk.



2. ábra: DTW algoritmus alapötletének sematikus rajza

A hasonlóságot a sorozatok közti távolságfüggvény segítségével definiáljuk. Minél kisebb a távolság a két sorozat között, annál hasonlóbak. Azonos logika mentén minél nagyobb ez a távolság, annál jobban különböznek egymástól. A távolságfüggvényt szokták költségfüggvénynek is nevezni. A cél tehát ennek a minimalizálása.

Az alapötlet a két idősor pontjai között valamilyen egymásnak való megfeleltetés meghatározása. Vagyis olyan (p_1, p_2) pontpárokat szeretnénk generálni, ahol p_1 az egyik, míg p_2 a másik idősor egy pontja, és a p_1, p_2 pontokat egymással feleltetjük meg. A konkrét megvalósítás a DTW algoritmus megvalósítása című fejezetben olvasható.

2.3.2 Távolságmértékek

Sok módszer, ahogy a DTW algoritmus is, végeredményben valamilyen klasszikus távolságmértéken alapszik, azt használja működéséhez. Az eredmények tekintetében ennek megválasztása is befolyással bírhat. A teljesség igénye nélkül szeretnék megemlíteni pár alkalmazott távolságszámítási módszert.

Az egyik ilyen, ami talán a legkézenfekvőbb az egyszerű Euklideszi-távolság. Sokszor alkalmazott szintén a Manhattan-távolság, melyet city block distance néven is gyakran említene. A klasszikusabb megközelítések mellett persze használnak más megoldásokat is, például a Mahalanobis-távolságot. [18]

2.4 Eredmények értékelése

Az egyes aláírás-hitelesítési rendszerek, módszerek pontosságát, helyességét szeretnénk egységes keretek közt megadni. Alapvető elvárásunk, hogy a rendszer jó döntéseket hozzon, vagyis az eredeti aláírásokat eredetinek, míg a hamisakat hamisnak ítélje. Sajnos azonban ez nem minden esetben teljesül. Jó megközelítés tehát, ha arra törekszünk, hogy minél nagyobb arányban hozzon helyes, és minél kisebb arányban hibás, helytelen döntéseket a rendszer.

A következőkben szeretném bemutatni azokat az általánosan használt hibaarányokat, melyeket a szakirodalom a legtöbb esetben alkalmaznak. Ezek az egységesen használt eredményességi mutatók lehetővé teszik a különböző kutatási eredmények összehasonlíthatóságát.

A korábbiaknak megfelelően mindig arra törekszünk, hogy ezek a mutatók a lehető legkisebb értékeket ériék el, ez jelenti azt, hogy a hitelesítés pontos, megbízható eredményeket szolgáltat.

2.4.1 False Rejection Rate (FRR)

Az automatikus aláírás-hitelesítés során kétféle hibatípus fordulhat elő. Az első ilyen az angol terminológiában Type I Error kifejezéssel hivatkozott, amikor egy eredeti aláírást tévesen elutasítunk. Ez az a hibás döntés, amikor egy eredeti aláírást, vagyis ami valóban az adott személytől származik, hamisnak ítélünk meg, azaz ennek tényét nem ismerjük el.

Az aláírás-hitelesítés során bekövetkezett ilyen típusú hibák arányát gyakran False Rejection Rate néven említik, jelölése α . Kiszámítási módja a következő képlet segítségével írható le:

$$\alpha = \frac{\text{tévesen elutasított eredeti aláírások száma}}{\text{összes eredeti aláírások száma}}$$

2.4.2 False Acceptance Rate (FAR)

A másik lehetséges hibatípusra az angol terminológia Type II Error megnevezéssel hivatkozik. Ez azt az esetet írja le, amikor egy hamis aláírást tévesen eredetinek ítélünk meg. Vagyis hibásan elismerjük a tényt, hogy az aláírás valóban az adott személytől származik, miközben valójában nem így van.

Az aláírás-hitelesítés során bekövetkezett ilyen típusú hibák arányát gyakran False Acceptance Rate néven említik, jelölése β . Kiszámítási módja a következő képlet segítségével írható le:

$$\beta = \frac{\text{tévesen elfogadott hamis aláírások száma}}{\text{összes hamis aláírások száma}}$$

2.4.3 Equal Error Rate (EER)

A fenti két hibarány között egyértelműen meghatározható, szoros kapcsolat van. Mindkét aránymutató valamilyen hasonlósági metrika alapján meghatározott határértéktől függ. Tegyük fel, hogy a hasonlóság mérésére olyan mérték szolgál, aminek köszönhetően a határértéknél kisebb értékkel bíró aláírásokat tekintjük eredetinek, míg az annál nagyobbal rendelkezőket hamisnak. A két mutató között egy a fordított arányossághoz hasonló kapcsolat figyelhető meg. A határérték növelésével FRR értéke csökkenthető, ezzel egyidejűleg azonban FAR értéke, ha változik – maradhat változatlan is –, akkor nőni fog. Hasonló logika mentén a határérték csökkentése FRR növekedését és FAR csökkenését eredményezheti.

A határérték optimális megválasztásával elérhető, hogy FRR és FAR egyenlők legyenek. Ez az arány az Equal Error Rate (EER), más néven Crossover Error Rate (CER). A cél ennek a hibaaránynak a csökkentése, amennyire csak lehetséges. Tehát FRR és FAR lehető legkisebb értékét szeretnénk egyidejűleg elérni.

Ez a mutató gyors lehetőséget biztosít különböző hitelesítőrendszerek megbízhatóságának összehasonlítására. Egy rendszert annál megbízhatóbbnak tekintünk, minél pontosabb. A pontosabb, megbízhatóbb rendszert az EER alacsonyabb értéke jelenti számunkra.

2.4.4 Average Error Rate (AER)

AER értékét, adott adatkészlet és határérték szerint számított, FAR és FRR átlagát véve számíthatjuk ki.

Azokban az esetekben, amikor EER pontosan nem meghatározható, de a különbség FAR és FRR között elég kicsi, az EER egy elfogadható közelítése AER. A munkám eredményét vizsgálva én is az Average Error Rate értékét vettem figyelembe.

2.5 Adatbázisok

Vannak publikált eredmények, melyek saját adatbázison (is) végezték a kiértékelést. Ez indokolt lehet például, ha valamilyen speciális környezetet feltételezve folyik a kutatás. Ilyen lehet, ha mobileszközök segítségével, azokon szeretnénk végezni az aláírás-hitelesítést. [19] A világ fejlődése, a mobileszközök hétköznapi széleskörű használata ezt az igényt abszolút alátámaszthatja.

A tapasztalat azonban azt mutatja, hogy a legtöbb esetben az összehasonlítható, egységes eredmények érdekében, vagy a saját adatbázis kialakítására való kapacitás hiánya miatt, de általában valamely nyilvánosan elérhető, híres adatbázis szolgáltatja a rendszerek tesztelésének alapját. Ilyenek például az Aláírás-hitelesítés jelentősége című fejezetben említett nemzetközi versenyeken használt adatbázisok. [20]

A használt adatbázisok kiválasztása is hatással lehet az adott módszer eredményeire. Előfordulhat például, hogy európai aláírásokra egészen nagy bizonyossággal helyesem működik egy rendszer, azonban a jelentősen eltérő valamely keleti országban felvett aláírásokra már messze nem hoz akkora sikert. [21] [22]

A különböző adatbázisok felépítése is fontos, ezért pontosan meghatározzák, hogy milyen formában történt az aláírások rögzítése, a hamisítványok profik-e (skilled) vagy más aláírók eredeti aláírásait használták (random). Fontos információ, hogy hány aláíró vett részt és hozzájuk hány eredeti és hamis aláírást rendeltek. Ezen felül, hogy pontosan milyen aláírástulajdonságokkal (X, Y, nyomás stb.) írhatjuk le az így kapott aláírásokat.

2.5.1 Signature Verification Competition 2004 (SVC2004)

Az egyik leghíresebb és azóta is számtalanszor használt SVC2004 adatbázis az első nemzetközi aláírás-hitelesítési versenyhez köthető. Mivel ez elég hosszúéletűnek bizonyult, és a munkám során én is ezt használtam, ezért ennek felépítését szeretném részletesebben bemutatni. [23]

Task1 és Task2 néven két, különböző aláírás-tulajdonságokat tartalmazó adatbázisból áll, miszerint a Task1-ben csak koordinátainformációkat tárolnak (időbélyeggel együtt természetesen), addig a Task2 aláírásaihoz egyéb plusz információk is társulnak. Mivel a Task1 tulajdonságai részhalmazát képezik Task2 által számontartott tulajdonsághalmaznak, és a munkám során én a Task2-vel dolgoztam, így utóbbinak a felépítését fogom részletezni.

Az aláírások rögzítése digitális tabletek segítségével történt. A résztvevő versenyzők rendelkezésére az összesen 100 közreműködő aláíró közül, a fejlesztés és tesztelés elősegítésére csak 40 aláíró aláírásait bocsátották. Minden aláíróhoz tartozik 20 eredeti aláírás és 20 profi hamisítvány, utóbbiak 5 másik résztvevő segítségével kerültek rögzítésre.

Minden egyes aláírás egy-egy külön (.TXT kiterjesztésű) szövegfájlban van tárolva. A fájlok elnevezési konvenciója UxSy a következők szerint. Az U, mint user után álló x egy szám, mely az aláíró azonosítója. Az S, mint signature után álló y szintén egy azonosító szám, mely az aláíró konkrét aláírását egyértelműsíti. A fentieknek megfelelően x és y értéke is 1 és 40 között fekszik, azonban y esetén az eredeti aláírások sorszáma minden esetben 1 és 20 közé esik, míg a hamisaké 21 és 40 közé.

Minden aláírás egyszerűen pontok egy sorozataként van reprezentálva. A fájlok első sora minden esetben egyetlen egész szám, ami az aláírást alkotó pontok számát jelenti. Ezután a fájlok minden egyes sora egy-egy pontnak felel meg, soronként a különböző tulajdonságok a következő sorrendben vannak feltüntetve:

- X-koordináta: kurzorpozíció az x-tengely mentén
- Y-koordináta: kurzorpozíció az y-tengely mentén
- Időbélyeg: rendszeridő, amikor a pont rögzítésre került
- Fej állapota: a fej aktuális függőleges pozíciója
(0 – toll felemelve, 1 – toll lent)
- Irányszög (azimuth): a toll óramutató járásával megegyező irányú elfordulása a Z-tengely körül
- Magasság (altitude): a toll és a pozitív Z-tengely által felfelé bezárt szöget határozza meg.
- Nyomás: normál nyomás korrigált helyzete

3 Használt megközelítésem

A célom a DTW alapú on-line aláírás-hitelesítés tanulmányozása volt, annak vizsgálata, hogy különböző előkészítő lépések, algoritmusok használata, vagy különböző aláírás-tulajdonságok figyelembe vétele hogyan befolyásolja az eredményeket. Ezzel szerettem volna olyan következtetéseket levonni, melyek másokat is segíthetnek ezek megválasztásában. Ezen felül pedig célom volt a vizsgálataim eredménye alapján ezeket úgy kombinálni, hogy azzal egy megbízható aláírás-hitelesítőt kapjak.

3.1 Alapmegközelítés

Alapmegközelítésként csupán a legegyszerűbb eszközök alkalmazását vettem. Ennek megfelelően nem végeztem semmilyen előfeldolgozó lépést, az adatokat tisztán abban a formában használtam, ahogy azt az SVC2004 adatbázisban rendelkezésre bocsátották. Az aláírások összehasonlítása során DTW algoritmus segítségével határoztam meg a köztük lévő távolságokat. A hitelesítés alapesetben mindig kizárólag az Y-koordináta alapján történt.

3.1.1 DTW algoritmus megvalósítása

Valójában ez egy dinamikus programozási feladat, ami során egy mátrixot kell kitöltenünk. A mátrix oszlopai és sorai az egyik, illetve másik idősor pontjait jelölik. Egy-egy cella tartalmazza a minimális össztávolságot, mely a két vizsgált vektor közt addig a pontig meghatározható. A táblázat legelső cellája (1,1) a két vektor első pontjai közt mért távolságot tartalmazza. Az első oszlop további mezőiben a mező által meghatározott két pont közti távolságnak és a felül szomszédos mező értékének az összege szerepel. Hasonlóképp az első sorban a balról szomszédos mező és a két pont közti távolság összege áll.

A mátrix minden további mezőjét három korábbi, szomszédos mező ismeretében tudjuk kiszámítani. Elsőként meghatározzuk a kitöltendő mező által meghatározott két pont közti távolságot. Ezután megvizsgáljuk, hogy melyik mezőhöz tartozik a legkisebb össztávolság azok közül, ahonnan az új, üres mező elérhető. Az üres mezőt az imént meghatározott két érték összegével töltjük ki.

A 3. ábra által használt elnevezések szerint tehát a D mező kitöltése során vesszük A , B és C mezők közül a legkisebb értékkel rendelkezőt, valamint kiszámoljuk a $(4,5)$ pontpár közti távolságot. D ezen két szám összegét kapja értékül.

	1	2	3	4	5	6	7	8	9
1									
2									
3				A	B				
4				C	D				
5									
6									
7									N

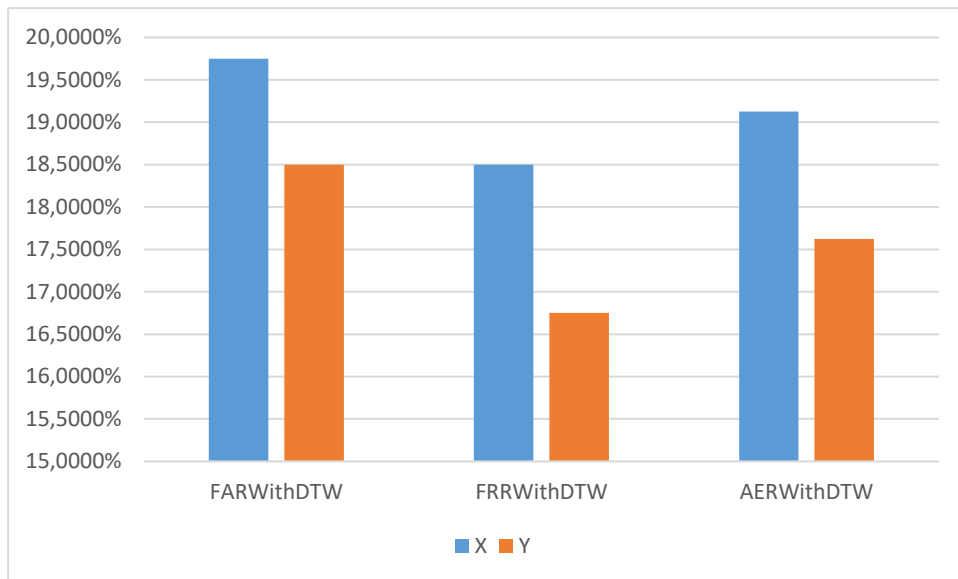
3. ábra: DTW költségmátrixa

A mátrix jobb alsó cellája, melyet a 3. ábra N betűvel jelöl, tartalmazza a DTW algoritmus eredményét. Ennek a mezőnek az értéke határozza meg, hogy a két vektor pontjai között mekkora az a minimális össztávolság, ami fennáll. Ezt az értéket használjuk a hasonlóság meghatározására. A mátrixot költségmátrixnak nevezzük a Dynamic Time Warping című fejezetben ismertetett névhasználati szokások miatt, és mivel távolságértékeket tartalmaz.

3.1.2 Vizsgált aláírás-tulajdonság kiválasztásának háttere

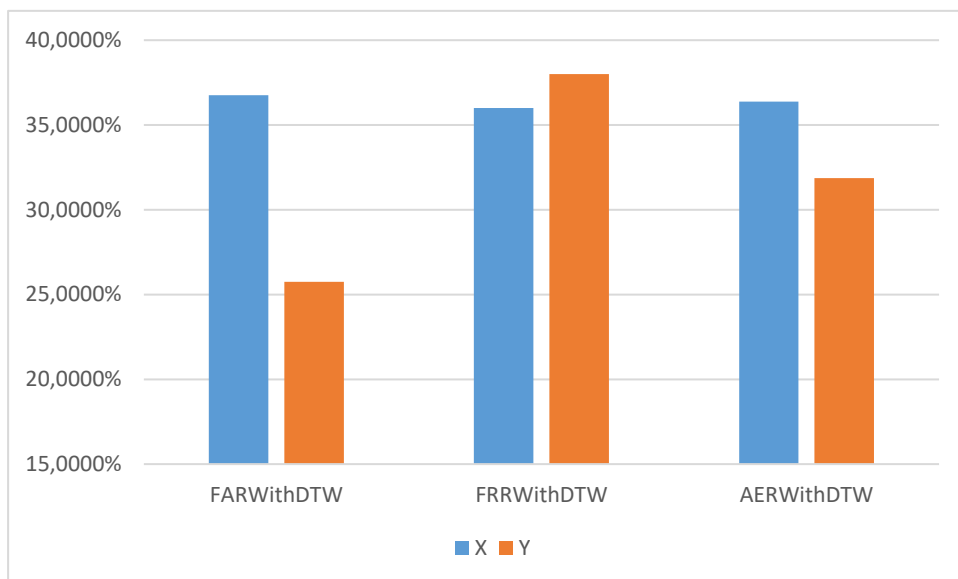
Az egyszerűsége törekedve mindenképp egyetlen aláírás-tulajdonság felhasználásával szerettem volna definiálni az alapesetet. A lehetőségek halmazát leszűkítettem az SVC2004 adatbázis Task1 részében is fellelhető koordinátainformációkra, azzal a céllal, hogy univerzálisan, sok más megoldással összemérhető eredményeket kapjak.

A választás végeredményben az X- és Y-koordináták között történt. A döntés meghozatalának elősegítésére méréseket végeztem, melyek során a távolságokat mindig DTW segítségével, a nyers aláírásokon határoztam meg. Ezeket felhasználva az Osztályozók típusai című fejezetben bemutatott mind a két osztályozási megközelítéssel megvizsgáltam, hogy az egyes koordinátainformációk alapján mennyire pontos döntések születnek. Az elért eredményeket a következő ábrák szemléltetik.



4. ábra: Minimális AER érték elérését célzó osztályozás eredményei (koordinátainformációk, DTW)

Az elméleti minimumot meghatározó osztályozó esetében egyaránt FAR, FRR és AER értékét figyelembe véve szemmel láthatóan alacsonyabb hibaarányokat eredményezett a hitelesítési folyamat az Y-koordináta vizsgálatával, mint az X esetében, ahogyan azt a 4. ábra is jól mutatja.



5. ábra: Automatikus osztályozás eredményei (koordinátainformációk, DTW)

Végeredményben az 5. ábra is ezt támasztja alá, hiszen az automatikus osztályozó eredményeként Y felhasználásával FAR majd 10%-kal alacsonyabb értéket vett fel, mint X alapján. AER értéke szintén az Y-koordináta alkalmazása felé billenti a mérleget. Az egyedüli pont, ahol az X-koordináta teljesített jobban, az az automatikus osztályozó által

elért FRR érték. Itt azonban a különbség a két koordináta eredményessége között a többihez képest szinte elhanyagolható.

Mindezek fényében a minél jobb eredmény elérésére törekedve úgy döntöttem, hogy az alapmegközelítem szerint vizsgált tulajdonság mindig az Y-koordináta lesz.

3.2 Osztályozók típusai

Az aláírások eredetiségéről osztályozók segítségével lehet dönteni. Minden osztályozó lényegében két fázisban működik. Az első, vagyis tanuló fázisban az osztályozó feladata, hogy megtalálja azt a határvonalat, mely az eredeti és hamis aláírások között húzódik. A konkrét megvalósítás során ez azt jelentette, hogy megadott referencia-aláírásokon, adott hasonlóság mérték használatával megkerestem annak egy olyan határértékét, aminek felhasználásával minél több helyes döntést lehet hozni. A második fázis a tesztelésről szól, ekkor kerül megállapításra, hogy a tényleges tesztadatokon milyen eredményesen használható az osztályozó. A tesztek során az osztályozó a besorolandó aláírás és a referencia-aláírások közt fennálló távolság, valamint a tanuló fázisban meghatározott határérték alapján hozza meg döntését. A klasszikus megközelítés, hogy amennyiben az új aláírás távolsága nem nagyobb a határértéknél, akkor eredetinek ítéljük, ellenkező esetben pedig hamisnak.

Én a tesztek során az osztályozó által nem csak egy bináris kimenetet adtam, hanem azt is, hogy mennyire biztos a döntésben az osztályozó. Ezt a besorolandó aláírás referenciáktól vett távolsága és a határérték közti különbséggel arányosan határoztam meg. Ha a tesztfázis bemeneteként kapott távolság épp a határértékkel egyenlő, akkor vagyunk 0,5 biztosságnál. Ez lineárisan nő vagy csökken, attól függően, hogy a távolság értéke a határértéknél mennyivel kisebb vagy nagyobb. A végső döntés tehát úgy születik meg, hogy amennyiben 0,5-nél nagyobb biztonsággal állítja az osztályozó, hogy az aláírás eredeti, akkor a döntés az, hogy eredeti. Amennyiben 0,5-nél kisebb biztonsággal mondja eredetinek, vagyis inkább hamisnak gondolja, akkor a döntés az lesz, hogy az aláírás hamis.

Az osztályozók általában két merőben különböző megközelítés egyikét alkalmazzák. Miszerint beszélhetünk aláírófüggő és aláírófüggetlen határértékekről. Előbbi esetben aláíróként egyéni, a többitől akár különböző határértéket állapít meg az osztályozó, csak az aláíróhoz tartozó aláírás referencia-aláírásai alapján. Utóbbi esetben ezzel szemben az egész adathalmazra egy globális határérték lesz érvényes, mely az

összes aláíróra megegyezik. Az irodalmakból egyértelműen kiderül, és szerintem intuitívan is következik, hogy aláírófüggő határértékekkel jobb eredmények érhetőek el. Ennek tudatában én minden esetben ily módon, aláírónként külön-külön meghatározott határértékekkel dolgoztam. [2] [7]

3.3 Minimális AER érték elérését célzó osztályozás

Ebben az esetben a cél adott eredeti és hamis tesztaláírásokon tesztelve a legjobb eredmény, vagyis a lehető legalacsonyabb AER érték elérése volt. Ezzel az osztályozóval tehát nem arra törekedtem, hogy valós viszonylatokban használható legyen, hanem arra, hogy megtalálja azt a legjobb eredményt, alsó korlátot, amit az aktuális paraméterezéssel elméleti szinten el lehetne érni. Ez az ismeret segíti az Automatikus osztályozás eredményeinek későbbi kiértékelését, hiszen az elméleti legjobb elérhető eredményhez viszonyítva jobban eldönthető, hogy mennyire lehetünk elégedettek.

3.3.1 Tanulási fázis

A nagy különbség, amiért éles helyzetben nem alkalmazható ez a fajta osztályozás, az az, hogy már a tanulási fázisban rendelkezésünkre áll, ismert számunkra minden tesztaláírás is. Továbbá természetesen rendelkezésre állnak eredeti referencia-aláírások is. Minden tesztaláírást a referenciáktól vett átlagos távolságával fogunk jellemezni.

Ennek megfelelően első lépésben minden tesztaláírásra kiszámoltam, majd átlagoltam a kapott referencia-aláírásoktól számított távolságát, így kaptam azokat az átlagtávolságokat, melyek alapján az osztályozást végeztem.

Második lépésben a határérték beállítása volt a feladat. Ennél ahogy már említettem az volt a cél, hogy az ismert tesztaláírásokon a lehető legpontosabban működjön. Ehhez az átlagtávolságokat növekvő sorrendbe állítottam, majd a határértéket elkezdtem a legkisebb értéktől kezdve léptetve állítgatni, és minden lépés során az aktuálisan meghatározott határértékkal elvégeztem a kiértékelést. A léptetések során mindig két egymást követő átlagtávolság között „fél úton” lévő értéket választottam. A kiértékelés során az egyes aláírásokról a korábban ismertetett módon (lásd Alapmegközelítés) bináris döntést hoztam a hozzá tartozó átlagtávolság és a határérték alapján. Ennek eredményeképp kaptam egy (határérték, FAR, FRR, AER) négyesekből

álló listát. Ebből választottam ki a legkisebb olyan határértéket, amelyik az elért minimális AER értékhez tartozik.

Az alapmegközelítés paramétereivel ilyen osztályozót alkalmazva 17,625%-os AER értéket sikerült elérnem.

3.3.2 Tesztfázis

Ezután a tesztfázisban egy-egy tesztaláírásról hoztam döntést az osztályozó segítségével. A kapott aláírásra ismét kiszámítottam, a korábbiakhoz hasonló módon, a referencia-aláírásoktól vett átlagos távolságát, majd ezt vettem össze a határértékkel. Ez alapján született meg a döntés, és adta az osztályozó eredményül, hogy milyen bizonyossággal tartja az adott aláírást eredetinek.

A tesztfázis során lényegében ugyanaz történik, mint a tanuló fázis minden egyes lépésének a végén, a kiértékelés során. A tesztaláírások már korábban is rendelkezésünkre álltak.

3.4 Automatikus osztályozás

Az automatikus osztályozás már olyan megközelítés, amely úgy működik, mint a valódi esetekben, nem feltételez semmilyen előismeretet. Első közelítésben az volt a célom, hogy kiderítsem a legegyszerűbb, intuitívan is következő eszközök segítségével meghatározott határértékek felhasználásával milyen eredményeket érhetünk el.

3.4.1 Tanulási fázis

Az osztályozó számára csak az eredeti referencia-aláírások állnak rendelkezésre, ennek megfelelően a tanulási fázis kizárólag ezekre támaszkodhat. Első lépésben tehát meghatároztam az eredeti aláírások közt mérhető átlagos távolságot. Ehhez minden referencia-aláírást összehasonlítottam a tőle különböző összes másikkal, ily módon páronként meghatároztam a távolságot, majd ezeket átlagoltam. Ezután minden aláíráshoz tartozott tehát egy átlagtávolság, mellyel azt jellemezni lehet.

Az egyszerűsége törekedve ezen értékek alapvető statisztikai jellemzőinek vizsgálatával határoztam meg a határértéket. Ennek során kizárólag az előbbieken megadott módon az aláírásokhoz rendelt távolságok átlagát és szórását számoltam ki, és alkalmaztam. Ennek eredményeképp a határértéket intuitív és empirikus módon úgy definiáltam, hogy a távolságok átlagát a szórás egy törtrészeivel emeltem meg.

Ilyen módon működő automatikus osztályozóval az alapmegközelítés paramétere mellett csupán 31,875%-os AER értéket sikerült elérni. Egyértelműen látszik, hogy ez az elméletileg elérhető eredményhez képest jócskán elmarad. Az osztályozó pontosságát a határérték meghatározásának valamilyen szofisztikáltabb módszerével tehát még lehet javítani. A céloom azonban első sorban nem ez volt, így ezzel nem foglalkoztam, ehelyett azt vizsgáltam, hogy a hitelesítési folyamat egyéb tényezői milyen hatással lehetnek erre az eredményre.

3.4.2 Tesztfázis

Az automatikus osztályozó csak a tesztfázis során kapja meg az eredeti és hamis tesztalíráásokat. A tesztelés menete innentől a korábban vázolttal (3.3.2. fejezet) megegyezik.

Első lépésként a tesztalírást összehasonlítom az összes referencia-alírással, ennek során kiszámolom a köztük lévő távolságot. Majd ezen értékek átlagaként kapom meg a tesztalíráás jellemzésére továbbiakban szolgáló átlagtávolság értékét.

Az így kapott távolságértéket a tanulási fázis során meghatározott határértékkel összevetve adja végül meg az osztályozó, hogy mennyire gondolja biztosnak, hogy a tesztalíráás eredeti.

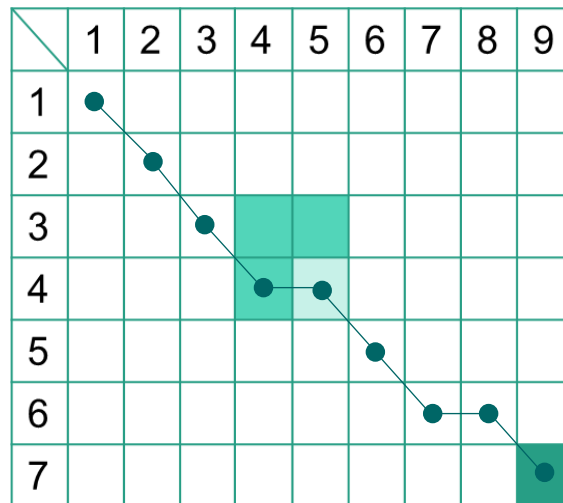
3.5 DTW és Warping Path Score kombinációja

Ezen megközelítés az Algoritmusok fejezetben említett néhány megoldáshoz hasonlóan a DTW valamely gyenge pontját hivatott kiküszöbölni, megerősíteni. Az alapgondolat az, hogy az eredeti DTW algoritmussal ellentétben, amely csak kizárólag külön-külön vizsgálja az egyes pontjait az összehasonlítandó vektoroknak, próbáljuk meg a pontok környezetét is figyelembe venni. Az algoritmus megvalósítását [24] alapján végeztem. Jelen dolgozatban erre a módszerre a továbbiakban fúziós megközelítés vagy FusedScore elnevezéseket is használom.

3.5.1 Warping Path Score lényege

A vázolt motiváció szerint kerül bevezetésre egy új érték, amit szintén használni lehet a hasonlóság megállapításához. Ennek lesz Warping Path Score a neve.

Kiszámításának első lépése egy hagyományos DTW algoritmus futtatása (lásd DTW algoritmus megvalósítása). A költségmátrix kitöltése után visszafejtéssel kerül meghatározásra az úgy nevezett Warping Path, ami a két vizsgált vektor elemeiből alkotott olyan pontpárok sorozata, amiken keresztül végig a minimális aggregált távolság mentén haladunk. Vagyis lényegében ez adja a DTW szerint leghasonlóbb pontok összerendelését. A 3. ábra N mezőjéből kiindulva, minden D mező esetén A , B és C közül az mentén haladunk tovább, amelyiknek értéke a legkisebb. Az első oszlop és sor mezői esetében értelemszerűen mindig csak egyetlen lehetőség van továbblépni. Végül elérünk az első sor első oszlopának mezőjéhez. Az út során pedig a Warping Path pontjai közé minden érintett mező után bekerültek az annak sora és oszlopa által meghatározott pontpárok.

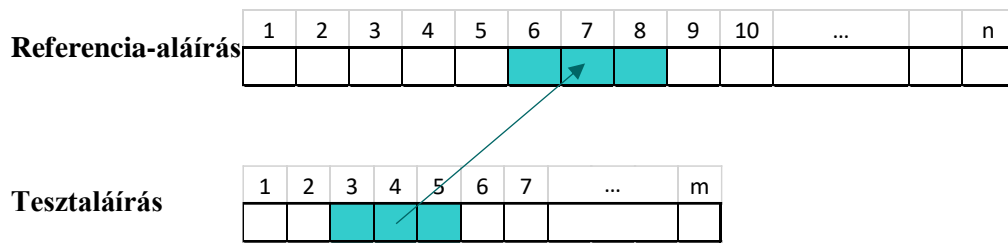


6. ábra: Warping Path egyszerű példája

Ezután a tesztalírák összes Warping Path által lefedett pontjához megkeressük a referencia-alírák azon pontját, amely az egyes pontok környezetét is vizsgálva a legvalószínűbb megfelelője lehet.

A környezet vizsgálatára úgy használjuk a DTW egy módosított változatát, hogy megadott szélességű ablakokat futtatunk végig a két vektoron. Néhány kisebb mérés eredménye alapján az ablakméretet 15 alíráspont szélesre állítottam, és a további méréseimet így végeztem. Az ablakok középső pontjához keressük mindig a másik vektorbeli megfelelőjét, még hozzá az ablakok távolsága alapján azt a pontot rendeljük hozzá, amelyik a legkisebb távolságra lévő ablak középső pontja a másik vektorban. Amennyiben két ablakot is talál az eljárás, mely a legkisebb távolságra fekszik, akkor a középső pontok távolsága lesz a mérvadó. Az ablakok távolságát úgy kapjuk meg, hogy

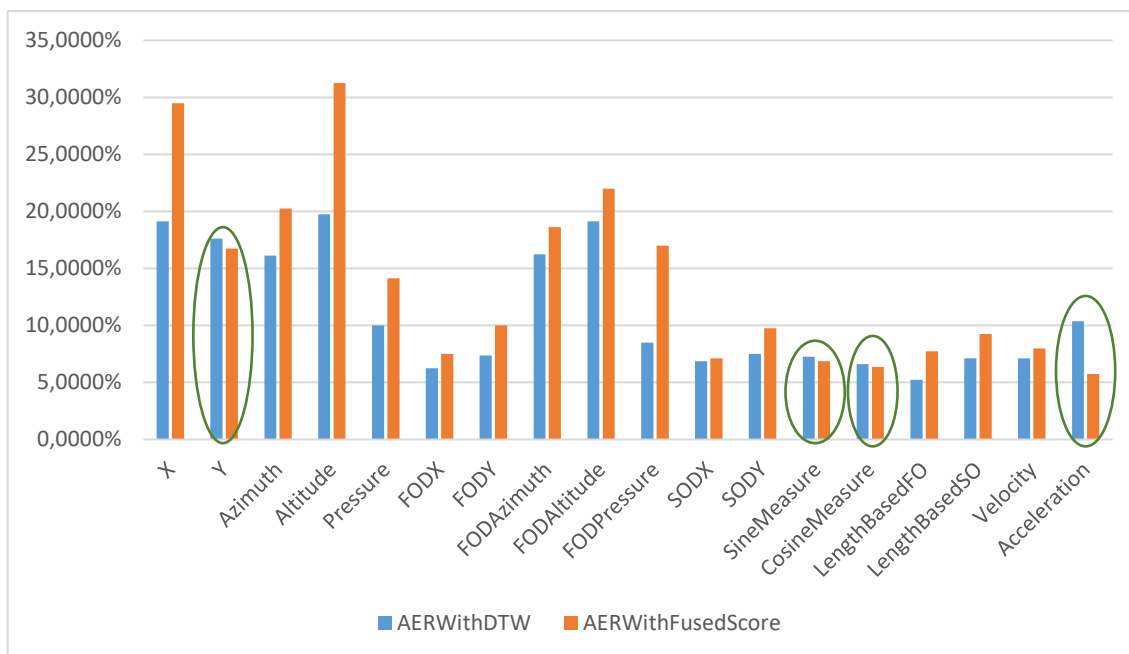
egyszerűen végigfutunk párhuzamosan a két ablak pontjain és a páronként vett DTW távolságaikat összegezzük.



7. ábra: Ablakos illesztés két alíráás pontjai közt

Ez az eljárás nem biztos, hogy ugyanazon pontokat rendeli a tesztalíráást meghatározó vektor Warping Path által fedett pontjaihoz a referencia pontjai közül, mint amiket a Warping Path tartalmaz. A tesztalíráás minél több pontja esetében rendeli hozzá ugyanazt, vagy nagyon közel eső pontokat a sima DTW és a vázolt ablakos módszer, annál valószínűbb, hogy valóban azok feleltethetők meg egymásnak. Ebből következően a Warping Path Score definiálása során a két módszer által meghatározott referencia pontok indexe közti különbség, valamint a tesztponttól vett távolságaik közti különbség összegzését vesszük. Az osztályozás során a DTW által meghatározott érték és a Warping Path Score összegével jellemezzük az egyes alíráásokat.

A módszer általam implementált megvalósítása a 8. ábra által jelzett eredményeket hozta a hagyományos DTW algoritmussal szemben. Maga a meg gondolás ugyan nagyon kecsegtető, a megvalósításom a nyers – azaz előfeldolgozási lépésen nem esett át – SVC2004 adatbázisbeli adatok felhasználásával, a tárolt és az azokból származtatott tulajdonságokon (bővebben az Alíráás-tulajdonságok fejezetben) nem aratott osztatlan sikert. Az új módszer sok tulajdonság esetén hasonló vagy rosszabb AER értéket eredményezett, mint a DTW, ám akadtak olyanok is, melyeken javulást hozott. A származtatott gyorsulás (acceleration) vizsgálatával például a DTW által elért 10,375%-os AER értéket 5,75%-ra, majdnem a felére sikerült csökkenteni.



8. ábra: DTW és DTW+WPScore módszerek összehasonlítása az elért AER értékek alapján (Minimális AER érték elérését célzó osztályozás)

3.6 Aláírás időbeli hosszának felhasználása

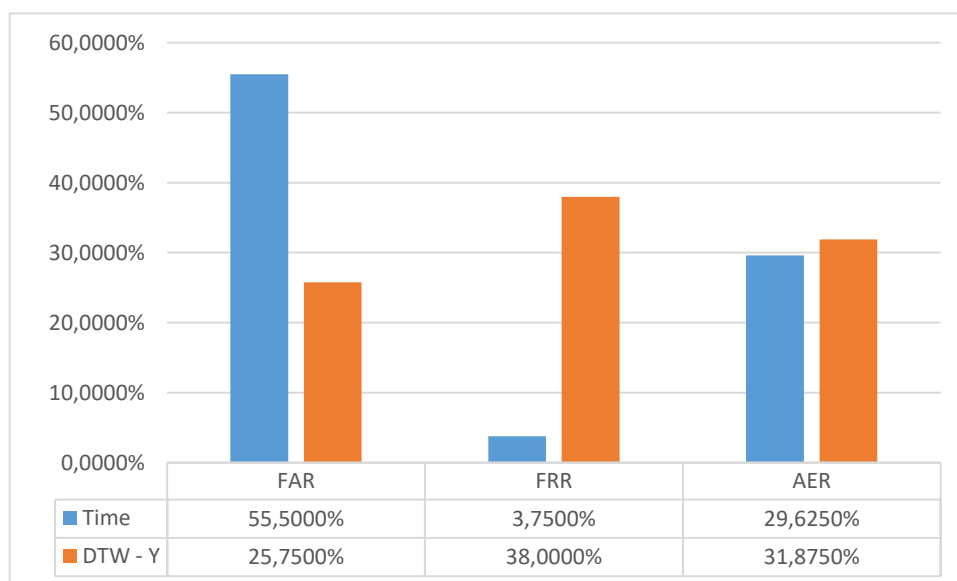
Az az egyszerű, de nagyszerű gondolat, hogy míg az eredeti aláírásokat valószínűleg hasonló idő alatt vetették „papírra”, addig a hamisítványok esetében ez gyaníthatóan lassabban történik. Értelemszerűnek tűnik, hogy egy pontos másolat elkészítése általában nagy körültekintést igényel, így több időt vehet igénybe. A gondolat tudományos megalapozottságáról a [25] 5. fejezetében olvasható.

Az aláírások időtartamát végeredményben tekinthetnénk származtatott aláírás-tulajdonságnak is, azonban mivel ez egy lényegileg különböző megközelítés a többi tulajdonság felhasználási módjához képest, ezért ezt kiemeltem azok közül. Ennek megfelelően a rendszeremben is másként kezelem, és ebben a dolgozatban is külön fejezetet szánok neki.

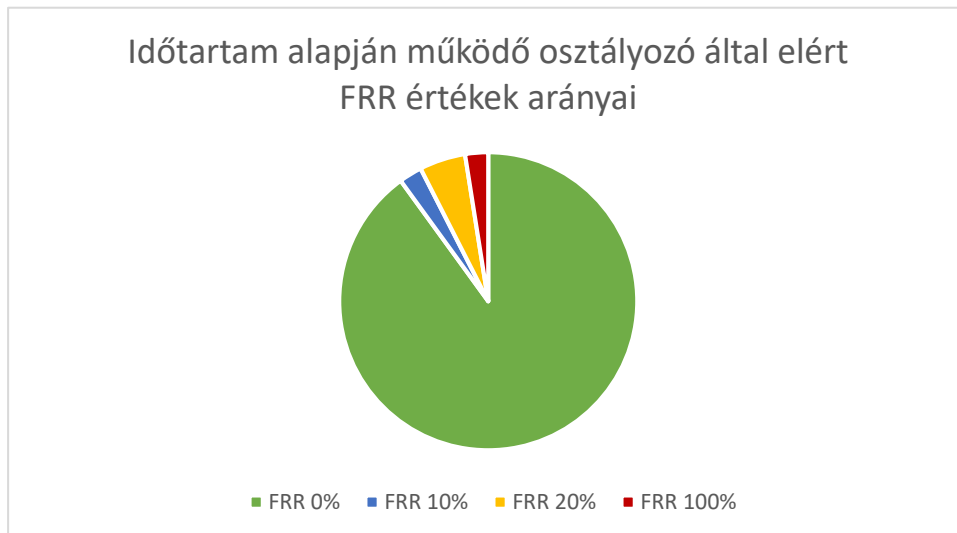
Első lépésben létrehoztam egy olyan automatikus osztályozót, amely csak az aláírások időtartama alapján hoz döntéseket. A tanulási fázis során a kapott referencia-aláírások mindegyikére kiszámítom annak az időtartamát a legnagyobb és legkisebb időbélyegének különbségként. A határérték megállapításához a referenciák között megfigyelhető legrövidebb és leghosszabb időtartamokat használtam fel oly módon, hogy a maximális időtartam értékét megemeltem a maximális és minimális mért időtartamok

közti különbség értékével. A tesztfázis során egyszerűen a kapott aláírásra kiszámított időtartam és a korábbiakban meghatározott határérték alapján született döntés. Ennek megfelelően az osztályozó eredményül egy $[0,1]$ intervallumba eső értéket adott, amely azt mutatja, hogy milyen bizonyossággal ítéli eredetinek az aláírást.

Az iménti működéssel bíró osztályozóval az SVC2004 adatbázison elért eredményeket a következő ábrák szemléltetik. Amint azt a 9. ábra eredményei is jól mutatják, az idő alapján történő szűrés gyakorlatban is jól működik. Tisztán az időtartam alapján működő osztályozóval (29,625%) az alapmegközelítéshez képest (31,875%) alacsonyabb AER értéket sikerült elérni. Az ábráról az is leolvasható, hogy noha az AER eredmények közel azonosak, az alapmegközelítés mégis abból a szempontból kiegyensúlyozottabb, hogy az AER érték EER durva közelítésének tekinthető. Az időalapú szűrésről ez nem mondható el, de ezzel nincs is semmi gond, hiszen a cél az volt, hogy idő alapján a hamisítványoktól meg tudjunk szabadulni.



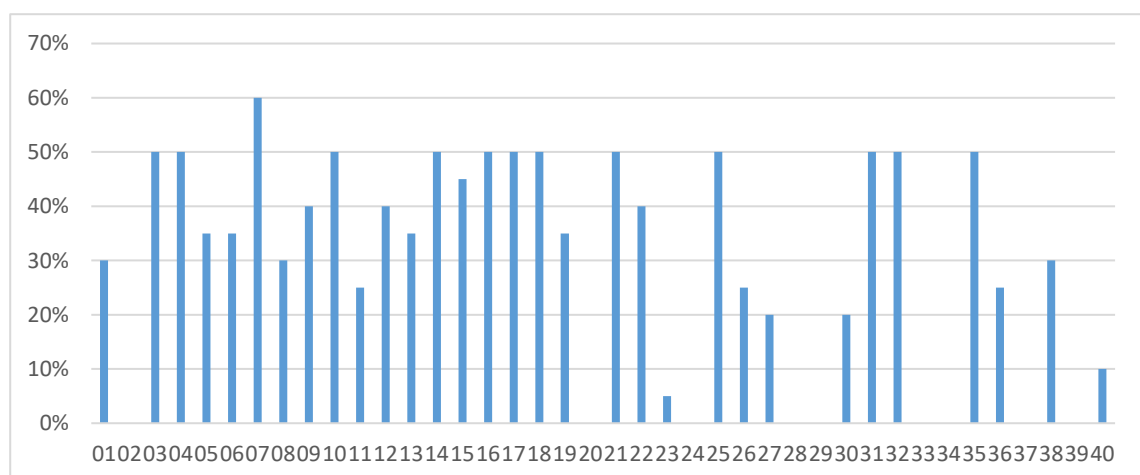
9. ábra: Az aláírás időbeli hossza alapján működő automatikus osztályozó és az alapmegközelítés által az SVC2004 adatbázis 40 aláíróján elért átlagos hibaarányai



10. ábra: Az aláírás időbeli hossza alapján működő automatikus osztályozó által az SVC2004 adatbázis 40 aláírójának aláírásain elért különböző átlagos FRR értékek arányai

Azt, hogy ez milyen jól sikerült a 10. ábra még inkább szemlélteti. A megközelítés az aláírók 90%-nál sikeresen működött abból a szempontból, hogy az időtartam alapján egyetlen eredeti aláírást sem utasítottunk el. A kiugró 100%-os értéknél, miszerint a 07 azonosítóval rendelkező aláíró összes eredeti tesztaláírását elutasítottuk, az a probléma, hogy a referenciaként kapott eredeti aláírások időbeli hosszuk szempontjából sokkal hasonlóbbak, mint a tesztaláírásokhoz képest.

Végül, de nem utolsó sorban a 11. ábra is alátámasztja az időszűrés létjogosultságát, hiszen jó néhány aláíró esetén 0%-os AER értékeket eredményezett, azaz 100% pontossággal hozott döntéseket a tesztaláírásokról.



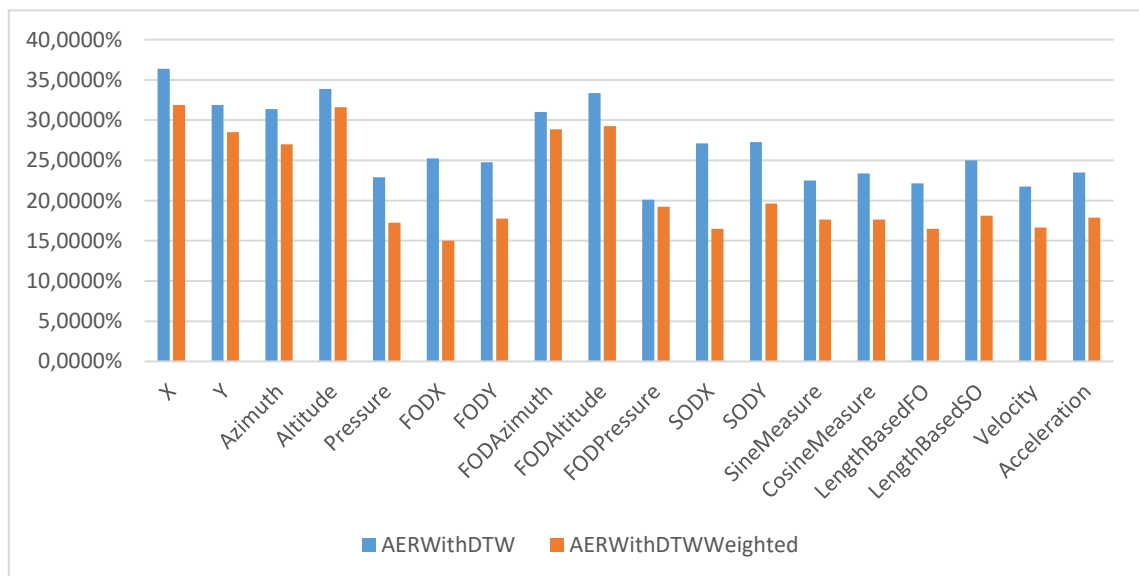
11. ábra: Az aláírás időbeli hossza alapján működő automatikus osztályozó által az SVC2004 adatbázis 40 aláírójának aláírásain elért átlagos AER értékei

3.6.1 Súlyozott, összetett osztályozó

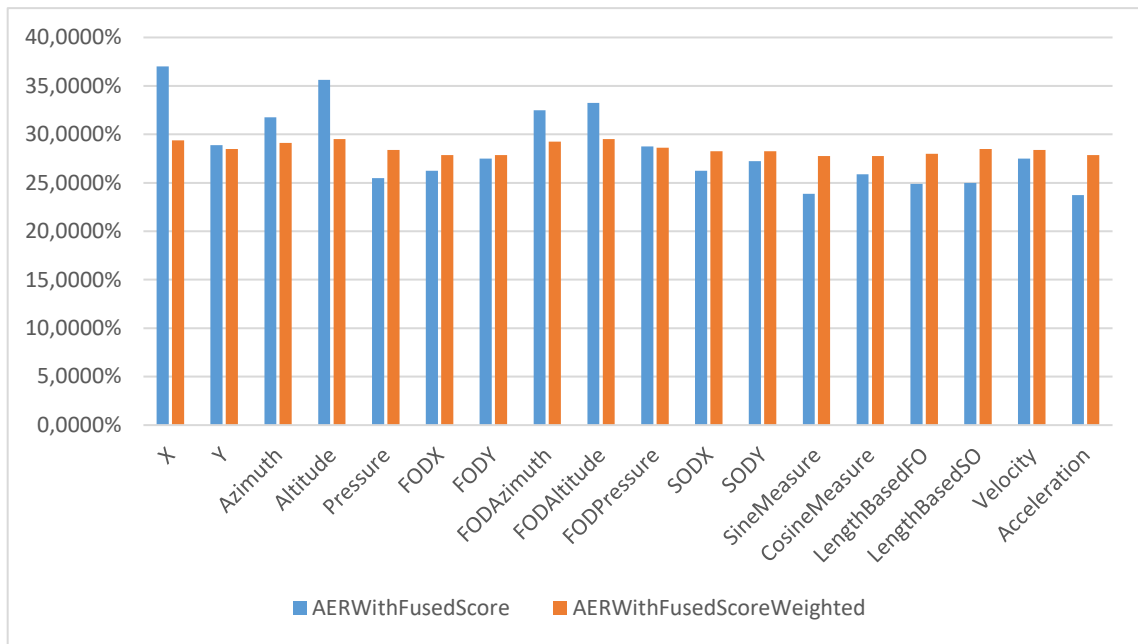
Most már, hogy tudjuk, hogy tisztán az aláírások időtartamát vizsgáló nagyon egyszerű osztályozóval is 30% körüli AER elérése lehetséges, jogosan merül fel, hogy az időszűrés alkalmazásával a meglévő algoritmusokat javítani próbáljuk. Ebből fakad a súlyozott, összetett osztályozó ötlete, mely nem csinál mást, mint összekombinálja az időbeli hossz alapján hozott döntést egy másik tetszőleges automatikus osztályozó döntésével. A súlyozottságát az adja, hogy megadható a két osztályozó döntését milyen súllyal vegyük figyelembe a végső ítélet meghozatala során. Ennek beállításával tulajdonképpen az fejezhető ki, hogy melyik osztályozóban bízunk meg jobban, s ezáltal melyiknek adunk többet a véleményére.

Az elképzelés igazolására méréseket végeztem, melyek során mind a sima DTW alapú, mind pedig a DTW + WPScore alapokon működő automatikus osztályozókat összekombináltam az időtartamot felhasználóval. Minden esetben 0,5-0,5 súlyozást kaptam mind az automatikus, mind pedig az idő alapján működő osztályozó. Egy átfogó kép érdekében az aláírás-tulajdonságok mindegyikének esetében elvégeztem a méréseket.

A DTW alapú automatikus osztályozó eredményeit a 12. ábra által szemléltetett módon sikerült befolyásolni az időszűréssel. Jól látszik, hogy kivétel nélkül, minden tulajdonság szerint sikerült javítani az eredményeken. A legszembetűnőbb változás FODX mellett figyelhető meg, ahol az eredeti 25,25%-ot több, mint 10%-kal sikerült csökkenteni, ezzel 15%-os AER értéket elérve.



12. ábra: Súlyozott, automatikus DTW és az aláírás időbeli hossza alapján működő osztályozó összetételével generált osztályozó eredményessége az SVC2004 adatbázis 40 aláíróján elért átlagos AER értékei alapján



13. ábra: Súlyozott, automatikus DTW+WPScore és az alírás időbeli hossza alapján működő osztályozó összetételével generált osztályozó eredményessége az SVC2004 adatbázis 40 aláíróján elért átlagos AER értékei alapján

A DTW+WPScore alapú, fúziós osztályozó automatikus változatának időtartam szerinti osztályozóval való kiegészítése már nem aratott ilyen egyértelmű eredményt, amint azt a 13. ábra is mutatja. Egyes esetekben a súlyozott osztályozó használata az AER értékek emelkedéséhez vezetett. Az idő figyelembevétele inkább elbizonytalanította az eredeti fúziós osztályozót, mintsem segítette volna annak döntéshozatalát. A különbség az eredeti és a kiegészített módszerek között kevésbé szembetűnő, aminek azért örülhetünk, mert így az eredményromlás sem olyan nagy arányú. Nagy örömünkre, azért néhány esetben javulás is történt, mely talán az X-koordináta esetén a legfeltűnőbb, ahol 37%-ról sikerült 29,375%-ra csökkenteni az AER értéket.

3.7 Aláírás-tulajdonságok

Ebben a fejezetben az aláírásokhoz rendelkezésre álló tulajdonságokat szeretném bemutatni, melyeknek a megbízhatóságát vizsgáltam a méréseim során, vagyis azt, hogy a különböző algoritmusok, előfeldolgozó lépések és osztályozó típusok megválasztása során mennyire hoznak kiegyensúlyozott eredményeket.

Alaptulajdonságoknak az általam használt SVC2004 adatbázisban (Task2) rögzített tulajdonságokat tekintem, melyeket a 2.5.1. fejezet részletesen ismertet.

3.7.1 Származtatott tulajdonságok

Származtatottnak tekintjük azokat az aláírás-tulajdonságokat, melyek közvetlenül nem állnak rendelkezésre az adatbázisban, azonban az alaptulajdonságokból levezethetők. Ezek megválasztására, amint azt a korábbiakban láthattuk rengeteg lehetőség van, én a következő néhány cikk [6] [24] [13] [26] meg gondolásaiból választottam ki egy számomra ígéretesnek tűnő halmazt.

Az általam használt származtatott tulajdonságok neveit, melyekkel jelen dolgozatban rájuk hivatkozom, valamint a hozzá tartozó definíciókat az 1. táblázat tartalmazza.

Rövid név	Teljes név	Definíció
FODX	First Order Differences of X	$\Delta x_r(i) = x(i+r) - x(i)$
FODY	First Order Differences of Y	$\Delta y_r(i) = y(i+r) - y(i)$
FODAzimuth	First Order Differences of Azimuth	$\Delta \phi_r(i) = \phi(i+r) - \phi(i)$
FODAltitude	First Order Differences of Altitude	$\Delta \theta_r(i) = \theta(i+r) - \theta(i)$
FODPressure	First Order Differences of Pressure	$\Delta p_r(i) = p(i+r) - p(i)$
SODX	Second Order Differences of X	$\Delta^2 x_r(i) = \Delta x_r(i+1) - \Delta x_r(i)$
SODY	Second Order Differences of Y	$\Delta^2 y_r(i) = \Delta y_r(i+1) - \Delta y_r(i)$
SineMeasure	Sine Measure	$\sin(\alpha_r(i)) = \frac{\Delta y_r(i)}{\sqrt{(\Delta x_r(i))^2 + (\Delta y_r(i))^2}}$
CosineMeasure	Cosine Measure	$\cos(\alpha_r(i)) = \frac{\Delta x_r(i)}{\sqrt{(\Delta x_r(i))^2 + (\Delta y_r(i))^2}}$
LengthBasedFO	Length Based Feature First Order	$l_r(i) = \sqrt{(\Delta x_r(i))^2 + (\Delta y_r(i))^2}$
LengthBasedSO	Length Based Feature Second Order	$\Delta l_r(i) = \sqrt{(\Delta^2 x_r(i))^2 + (\Delta^2 y_r(i))^2}$
Velocity	Velocity	$v(i) = \frac{\sqrt{(\Delta x_r(i))^2 + (\Delta y_r(i))^2}}{\Delta t_r(i)}$
Acceleration	Acceleration	$a(i) = \frac{\Delta v_r(i)}{\Delta t_r(i)}$

1. táblázat: Általam használt származtatott tulajdonságok

$i = 1, 2, 3 \dots n$, ahol n az aláírás pontjainak számát jelöli. Az eltolási paraméter (r) azt mutatja meg, hogy milyen indextávolságra lévő pontok között végezzük a számításokat. A definíciós szabályokban az alaptulajdonságok jelölése a következő:

- X-koordináta: x
- Y-koordináta: y
- Irányszög (azimuth): ϕ
- Magasság (altitude): θ
- Nyomás: p
- Időbélyeg: t

Tulajdonság	AER (DTW)	AER (DTW+WPScore)
X	19,1250%	29,5000%
Y	17,6250%	16,7500%
Azimuth	16,1250%	20,2500%
Altitude	19,7500%	31,2500%
Pressure	10,0000%	14,1250%
FODX	6,2500%	7,5000%
FODY	7,3750%	10,0000%
FODAzimuth	16,2500%	18,6250%
FODAltitude	19,1250%	22,0000%
FODPressure	8,5000%	17,0000%
SODX	6,8750%	7,1250%
SODY	7,5000%	9,7500%
SineMeasure	7,2500%	6,8750%
CosineMeasure	6,6250%	6,3750%
LengthBasedFO	5,2500%	7,7500%
LengthBasedSO	7,1250%	9,2500%
Velocity	7,1250%	8,0000%
Acceleration	10,3750%	5,7500%

2. táblázat: Minimális AER érték elérését célzó osztályozás eredményei a különböző tulajdonságok és algoritmusok megválasztása mellett

A 2. táblázat jól szemlélteti, hogy a származtatott tulajdonságok alkalmazása valóban hasznos. DTW használata mellett az esetek túlnyomó többségében a származtatott tulajdonságok vizsgálatával az elért AER értékek kevesebb, mint a felére csökkennek az alaptulajdonságokon elért átlaghoz képest. Ugyanez elmondható a másik algoritmussal végzett mérési eredményekről is. Ez jól mutatja tehát, hogy a származtatott tulajdonságok több algoritmus alkalmazása mellett is stabilan jobb eredményeket hoznak.

Érdekes továbbá megfigyelni, hogy a hasonló tendencia ellenére a legjobb eredményeket elérő tulajdonságok az algoritmus megválasztásától függően különbözőek lehetnek. Míg az alapmegközelítésben használt DTW a LengthBasedFO, FODX, CosineMeasure és SODX tulajdonságok mellett érte el a legalacsonyabb AER értékeket, addig a DTW+WPScore megközelítés alkalmazásával az Acceleration, CosineMeasure, SineMeasure és SODX tulajdonságok végeztek az első négy helyen. Látszik, hogy van átfedés a legjobb négy listái között, mégsem teljesen azonosak.

Egyértelműen látszik továbbá, hogy az alaptulajdonságok közül a nyomás a legjobban használható, hiszen mindkét esetben ez hozta közülük a legjobb eredményeket. A legjobb származtatott tulajdonságok azonban még ezeket az értéket is durván a felére

csökkentették. Így DTW esetén a nyomás által elért 10%-os elméleti alsó korlátot a LengthBasedFO tulajdonság 5,25%-ra tudta levinni, a fúziós megközelítés esetén pedig a nyomás 14,125%-a 5,75%-ra volt csökkenthető az Acceleration tulajdonság vizsgálatával.

Az eredmények azt is megmutatják, hogy a legkevésbé használható tulajdonságok az X-koordináta, a magasság (Altitude) és annak első rendű különbségei (FODAltitude). Mind a két esetben ez a három tulajdonság volt a lista alján az elért alacsony AER értékeket tekintve. A magasság rossz jellemzőerejét az is mutatja, hogy az ebből származtatott tulajdonsággal sem sikerül jelentős javulásokat elérni, míg X hiába ért el rossz eredményeket, az abból származtatott tulajdonságok a ranglista élén szerepelnek.

A származtatott tulajdonságokkal elért 5-8% már nagyságrendileg összemérhető az említett irodalmakból ismeretes 0,5-4%-kal.

3.7.2 Tulajdonságok kombinációi

A korábbiakban bemutatottnak megfelelően érdekes lehet, hogy a különböző tulajdonságok valamilyen kombinációjának alapján végzett hitelesítés milyen eredményeket hoz, javíthat-e a tulajdonságok egyenként elért eredményeihez képest.

Az általam használt és bemutatott mind a két algoritmus, mivel alapjaiban tekintve megegyeznek, így képesek kezelni a tulajdonság-halmazok együttes vizsgálatát. Ennek során az aláírások egyes pontjait nem csupán egy adott tulajdonsághoz rendelt egy-egy értékkel jellemezzük, hanem mindegyikhez egy-egy vektort rendelünk. Ezek a vektorok meghatározott sorrendben tartalmazzák a vizsgált tulajdonság-halmaz egyes elemeihez rögzített értékeit az aláírásnak. A számítások során az aláírások közti távolságának meghatározása közben mind a két algoritmus a pontokhoz rendelt vektorok egymástól való távolságát használja fel ebben az esetben.

A méréseim során nem teszteltem a bemutatott tulajdonságok minden lehetséges halmazkombinációjának hatását, csupán néhány ígéretes összeállítást. Vizsgáltam tulajdonságpárok által az elérhető AER értékekre gyakorolt esetleges pozitív hatásokat.

Ezen párokat, a két algoritmusra külön-külön, a következőképpen határoztam meg. Vettem az adott algoritmus mellett a legjobb négy eredményt szolgáltató tulajdonságot, mivel ezek bizonyulnak a legígéretesebbnek. Ezen felül hozzáadtam még a listához az általam használt alapmegközelítés során alkalmazott Y-koordinátát, az előző

mérések alapján legpontosabbnak bizonyuló alaptulajdonságként a nyomást, valamint a másik algoritmus alkalmazásával legjobb eredményt produkáló tulajdonságot. Ezután az így kapott hét tulajdonságból előállítható összes tulajdonságpárra végeztem méréseket az elérhető minimális AER érték meghatározására. Az automatikus osztályozókkal végzett futtatások során csak a négy legjobb eredményt hozó tulajdonság és a nyomás által alkotott halmazból képzett párokat vizsgáltam. Az eredményeket a következő táblázatok szemléltetik.

	<i>FODX</i>	<i>CosineMeasure</i>	<i>SODX</i>	<i>Pressure</i>	<i>Acceleration</i>	<i>Y</i>
<i>LengthBasedFO</i>	3,375%	5,250%	3,375%	5,375%	5,250%	12,500%
<i>FODX</i>	–	6,250%	5,375%	6,250%	6,250%	12,875%
<i>CosineMeasure</i>	–	–	6,875%	3,750%	4,750%	17,625%
<i>SODX</i>	–	–	–	6,875%	6,875%	15,625%
<i>Pressure</i>	–	–	–	–	6,125%	17,625%
<i>Acceleration</i>	–	–	–	–	–	17,625%

3. táblázat: Minimális AER elérését célzó, DTW alapú osztályozó eredményeinek összehasonlítása a legjobb tulajdonág és a tulajdonságpárok között

A 3. táblázat megmutatja számunkra, hogy ez a tulajdonságok kombinációjának használata nagyon hatékony tud lenni. A DTW által elért legjobb elméleti AER minimum a *LengthBasedFO* tulajdonság vizsgálata mellett 5,25% volt. Ezt majd 2%-kal sikerült csökkenteni azáltal, hogy másik tulajdonsággal kombináltuk. Az így elért 3,375%-os eredmény az eddigi mérések alatt megfigyelhető legjobb eredmény, mely már abszolút az irodalmi 0,5-4%-kal összhangban van.

Meglepetést okozhat, hogy a párok által elért AER értékek nem feltétlenül az egyéni eredmények sorrendjének megfelelően alakulnak. Vagyis például az első és a harmadik legjobb tulajdonságokból alkotott párnál sokkalta jobb eredmény érhető el az első és a negyedik kombinációjával.

Külön érdekesség, hogy a nyomást, mint alaptulajdonságot is tartalmazó egyik pár a második legjobb eredményt tudta produkálni, mely jelentősen javít a legjobb egyéni eredményen.

	<i>CosineMeasure</i>	<i>SineMeasure</i>	<i>SODX</i>	<i>LengthBasedFO</i>	<i>Pressure</i>	<i>Y</i>
<i>Acceleration</i>	5,875%	5,250%	8,125%	7,750%	6,625%	17,125%
<i>CosineMeasure</i>	–	7,750%	8,000%	7,750%	7,125%	16,750%
<i>SineMeasure</i>	–	–	7,500%	7,625%	6,125%	16,625%
<i>SODX</i>	–	–	–	5,625%	7,625%	13,875%
<i>LengthBasedFO</i>	–	–	–	–	7,875%	11,625%
<i>Pressure</i>	–	–	–	–	–	16,750%

4. táblázat: Minimális AER elérését célzó, DTW+WPScore alapú osztályozó eredményeinek összehasonlítása a legjobb tulajdonág és a tulajdonságpárok között

A 4. táblázat a fúziós megközelítés eredményeit demonstrálja, melyből a korábbiakhoz hasonló következtetéseket vonhatunk le. Érdekesség, hogy bár ebben az esetben is sikerült két pár alkalmazásával is az egyéni legjobb AER érték alá menni, azonban messze nem olyan nagy mértékben, mint a hagyományos DTW mellett. A nyomás jó meghatározó képessége itt is megmutatkozik, hiszen a legjobb hat páros eredmény felénél a nyomást vizsgáltuk valamilyen származtatott tulajdonság mellett.

	<i>Velocity</i>	<i>LengthBasedFO</i>	<i>SineMeasure</i>	<i>Pressure</i>
<i>FODPressure</i>	21,375%	22,125%	24,000%	21,625%
<i>Velocity</i>	–	21,875%	21,375%	20,750%
<i>LengthBasedFO</i>	–	–	22,000%	22,125%
<i>SineMeasure</i>	–	–	–	23,750%

5. táblázat: Automatikus DTW alapú osztályozó eredményeinek összehasonlítása a legjobb tulajdonág és a tulajdonságpárok között

	<i>SineMeasure</i>	<i>LengthBasedFO</i>	<i>LengthBasedSO</i>	<i>Pressure</i>
<i>Acceleration</i>	21,375%	25,000%	25,375%	25,375%
<i>SineMeasure</i>	–	24,750%	25,125%	22,625%
<i>LengthBasedFO</i>	–	–	23,875%	25,125%
<i>LengthBasedSO</i>	–	–	–	25,125%

6. táblázat: Automatikus DTW+WPScore alapú osztályozó eredményeinek összehasonlítása a legjobb tulajdonág és a tulajdonságpárok között

Az automatikus osztályozóknál pont ellentétes viselkedést figyelhetünk meg, ahogyan azt az 5. táblázat és 6. táblázat is mutatja. A korábbiakkal ellentétben DTW használatánál egyáltalán nem sikerült javítani a legjobb tulajdonság egyéni eredményén,

viszont a fúziós megközelítés esetén 2,5%-kal jobb eredményt sikerült elérni. Azt azonban, hogy a nyomással érdemes lehet foglalkozni, azt ez a két táblázat is alátámasztja.

3.8 Előfeldolgozás használata

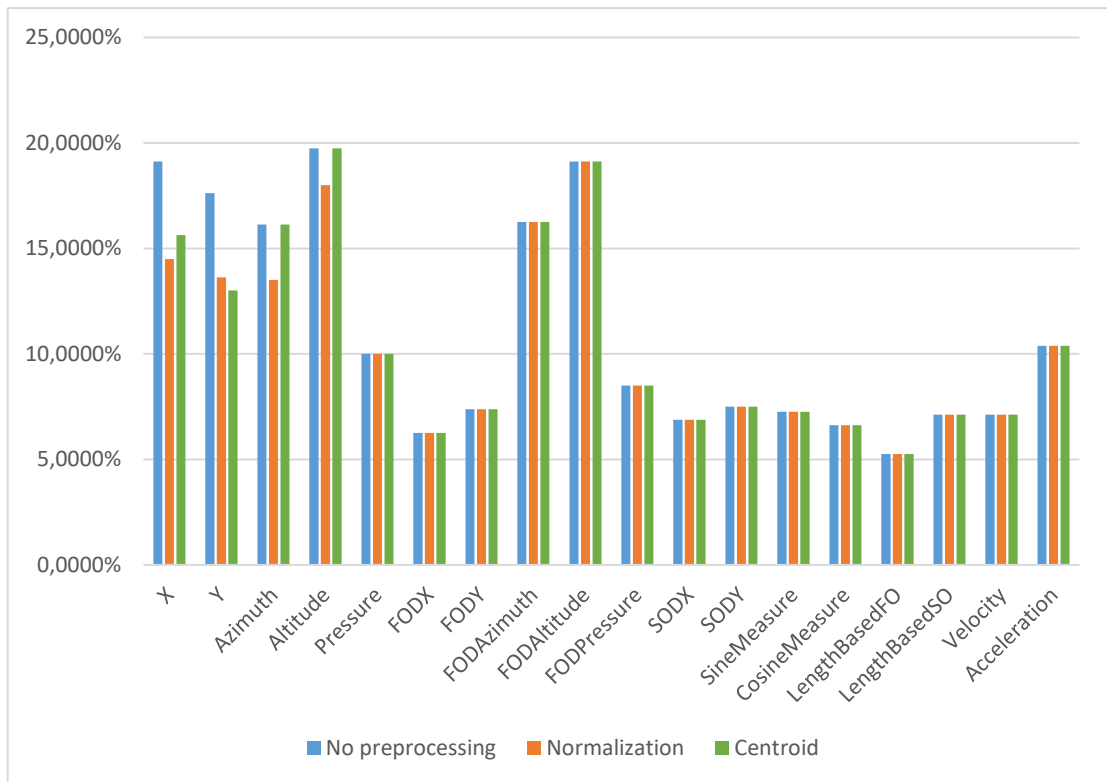
A számtalan lehetőség közül kettőt választottam, melyeknek az előfeldolgozás hiányához és az egymáshoz viszonyított eredményeit vizsgáltam. Ez a két egymást kizáróan használható módszer pedig nem más, mint a normalizálás és a súlypontba igazítás. Első megközelítésben ezeket a lépéseket a használt tulajdonság-halmaz definiálása után hajtottam végre, vagyis származtatott értékek esetében a már kiszámított értékeken végeztem el ezeket a transzformációkat.

A normalizálás célja, hogy a különböző tulajdonságok értékeit egymással összemérhetővé tegye. Ennek elérése érdekében minden a $[0;1]$ tartományba kerül leképzésre.

A súlypontba igazítás lényege, hogy az egymáshoz képest elcsúszott aláírásokat összehúzza, ezzel csökkentve az ilyenfajta torzulásból adódó esetleges eredményromlást. Az egyes aláírásokra meg kell határozni a súlypontjukat, majd a súlypontjaikat kell egymás fölé tolni. Ez az eljárás várhatóan a koordinátainformációkra van igazán hatással az alaptulajdonságok közül.

Mindkét előkészítési technikát alkalmaztam mind a két algoritmuson alapú mind a két osztályozó típus mellett, s erről méréseket végeztem, melyek eredményeit a következő néhány diagram szemlélteti.

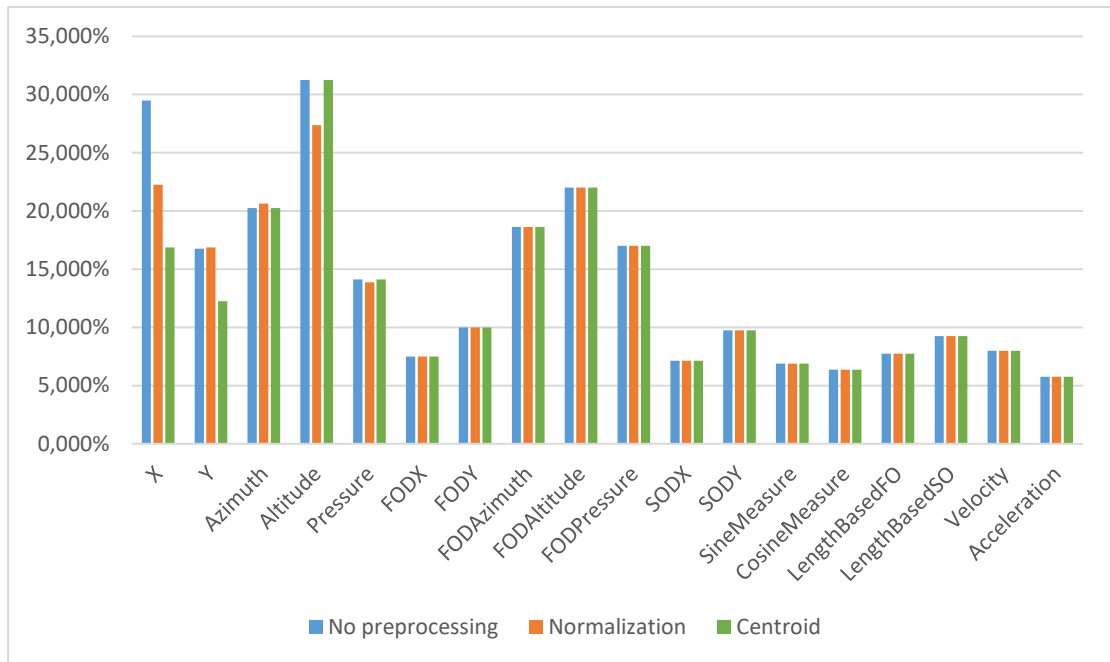
A 14. ábra jól mutatja, hogy a súlypontba igazítás az X és Y tulajdonságok vizsgálata esetén eredményez csak változásokat, ott viszont jelentős javulást mutat. X esetén a DTW alapú osztályozó által meghatározott elérhető minimális AER értéket 3,5%-kal, míg Y vizsgálata mellett 4,625%-kal csökkenteni a súlypontba igazítás segítségével. A normalizálás a koordinátainformációk alapján hasonlóan jó eredményeket hozott amellett, hogy a többi alaptulajdonság esetén is átlagosan majd 1,5%-os javulást mutatott.



14. ábra: Előfeldolgozó lépések hatása a minimális AER elérését célzó, DTW alapú osztályozó eredményeire

A 15. ábra hasonló tendenciákat mutat a DTW+WPScore alapú, azonos típusú osztályozó esetén is. A súlypontba igazítás itt is csak X és Y mellett elért eredményeket befolyásolta, de X mentén még nagyobb, több mint 12,5%-os csökkenést okozott. Érdekes megfigyelni, hogy a normalizálás bevezetésével a fúziós megközelítéssel Y vizsgálatával nem sikerült növelni a hatékonyságot.

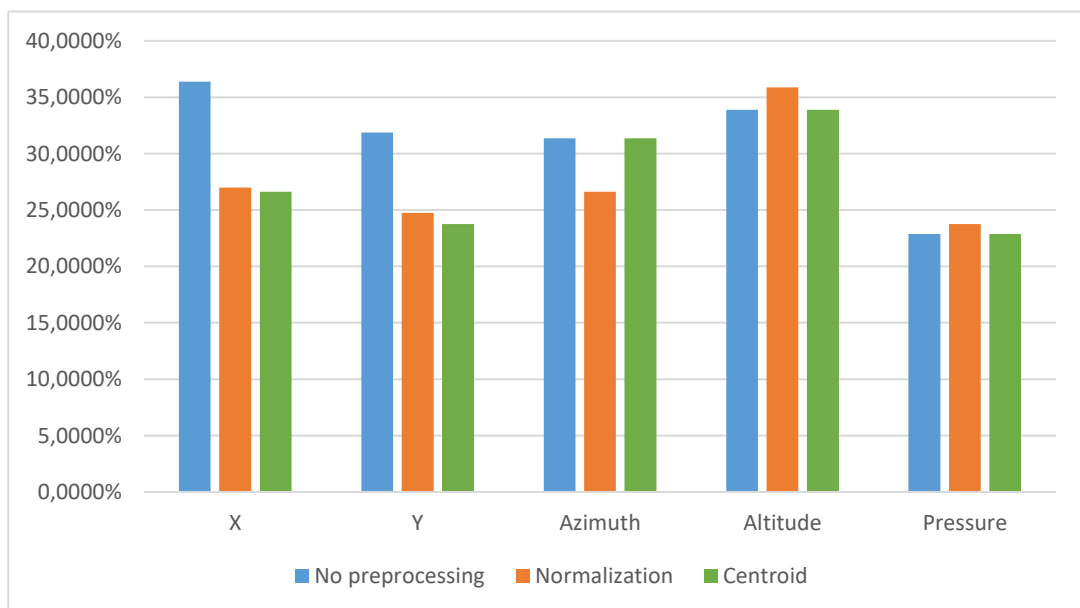
A két ábra nagyon jól szemlélteti mind e mellett, hogy a származtatott tulajdonságaink milyen jól használhatóak. Az ezekkel elért eredményeken egyik előfeldolgozó módszer sem tudott változtatni, ami azt mutatja, hogy ezeknek az előállítására már szépen eliminálta azokat a zavarokat, melyek miatt az előfeldolgozó lépések bevezetése értelmet kapott.



15. ábra: Előfeldolgozó lépések hatása a minimális AER elérését célzó, DTW+WPScore alapú osztályozó eredményeire

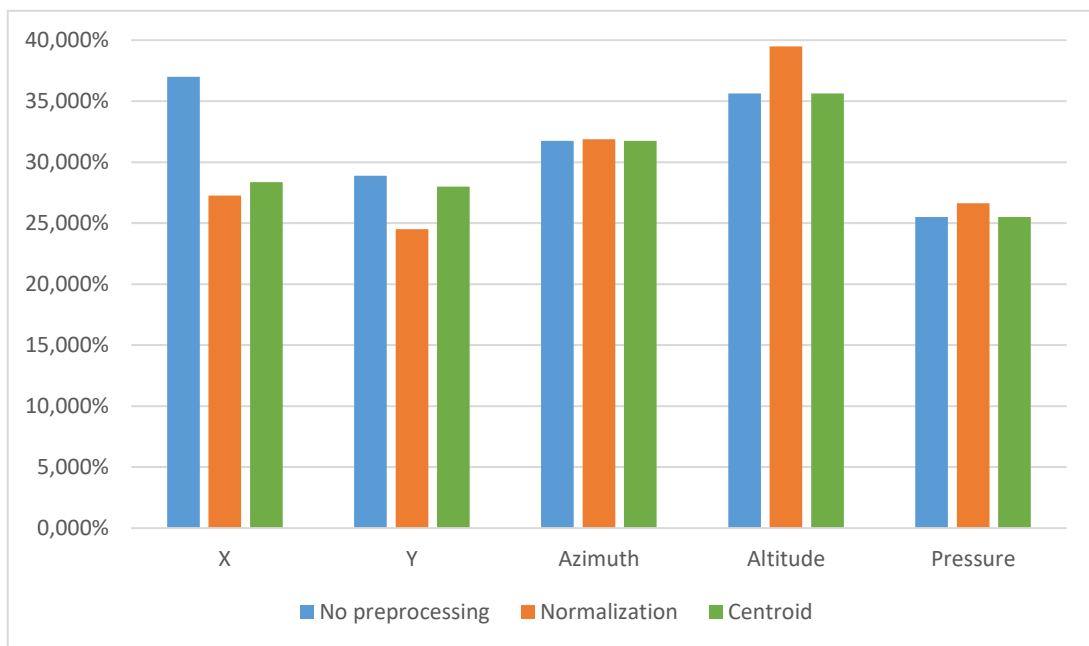
A mérések alapján a származtatott tulajdonságokra gyakorolt hatás hiányában az automatikus osztályozókat már csak az alaptulajdonságok szerint futtattam, és a következő ábrák által mutatott eredményeket kaptam.

A 16. ábra eredményei összhangban vannak a korábbiakkal, a DTW alapú osztályozó automatikus változata esetén is a súlypontba igazítás hasonló hatása figyelhető meg. Szembetűnő különbség azonban, hogy az Altitude tulajdonság mellett szemmel láthatóan ront az eredményeken a normalizálás.



16. ábra: Előfeldolgozó lépések hatása a DTW alapú automatikus osztályozó eredményeire

A 17. ábra már több különbséget is mutat a DTW+WPScore alapokon működő két osztályozótípus között. Míg a korábbiakban a koordinátainformációk szerint egyértelműen a súlypontba igazítással lehetett elérni az alacsonyabb elméleti minimális AER értéket, addig a gyakorlatban működő automatikus osztályozás esetében ez pont fordítottnak látszik. X és Y alapján is a normalizálás lesz hatékonyabb. Az a tendencia azonban továbbra is fennáll, sőt növekedett, hogy az azonos paraméterekkel rendelkező elméletileg elérhető legkisebb AER-t kereső osztályozóval ellentétben az automatikus megfelelője az Altitude tulajdonság vizsgálatával, nemhogy nagyobb javulást eredményezne, még határozottan ront is az elért AER értékeken.



17. ábra: Előfeldolgozó lépések hatása a DTW+WPScore alapú automatikus osztályozó eredményeire

4 Tesztelést segítő alkalmazás

A kutatásom során a méréseim elvégzésére, egy a tesztelési szempontok szerint definiált funkcionalitással bíró asztali alkalmazást fejlesztettem. Ezt szeretném nagyon röviden bemutatni. A munkám során a SigStat [27] aláírás-hitelesítést támogató keretrendszert használtam, melynek a fejlesztői közt vagyok én is. Az elkészült alkalmazás két fő funkcióval rendelkezik.

Az első az új módszerek korai fázisában nyújt segítséget azzal, hogy lehetővé teszi a különböző implementált algoritmusok által számított távolságértékek összehasonlítását, vizsgálatát. A másik funkció, már az egész hitelesítési folyamat, különböző osztályozók tesztelésére szolgál.

4.1 Implementált algoritmusok eredményeinek vizsgálata

Lehetőség van beállítani a számítások során használt tulajdonság-halmazt, ami szerint a távolságokat mérni fogjuk. Meg kell adni tovább, hogy melyik két aláírást szeretnénk összehasonlítani. Egy aláírást az aláíró azonosítója és az aláíróhoz tartozó aláírások között egyedi aláírás-azonosító határozza meg. Alapvetően ugyanazon aláíró két aláíráspéldányának megadását támogatja kényelmesen a szoftver, de szükség esetén lehetőség van különböző aláírók kézjegyei közti távolság meghatározására is.

4.2 Osztályozások eredményességének vizsgálata

A teljes hitelesítési folyamat tesztelésére szolgáló funkciókat is megvalósítottam. A keretrendszer által biztosított benchmark segítségével a konfigurációnak megfelelő osztályozó eredményeit lehet emberi szem számára olvasható, átlátható formában előállítani. A tesztelés és hibakeresés megkönnyítése céljából nem csupán a végeredményeket, hanem köztes részeredményeket is rendelkezésre bocsát az alkalmazás.

4.2.1 Konfigurációs lehetőségek

Az összes vázolt különböző megközelítés és egymást ki nem záró kombinációinak megadására van mód. Meg lehet adni a kívánt előkészítő lépést, az osztályozó típusát, valamint az annak alapját képező algoritmust. Továbbá az osztályozók tetszőlegesen megadott tulajdonság-halmazokon futtathatók, melyből előre többet is

meghatározhatunk, így lehetőséget biztosítva átfogó, nagyobb volumenű mérések kivitelezésére.

4.2.2 Osztályozás kimenete

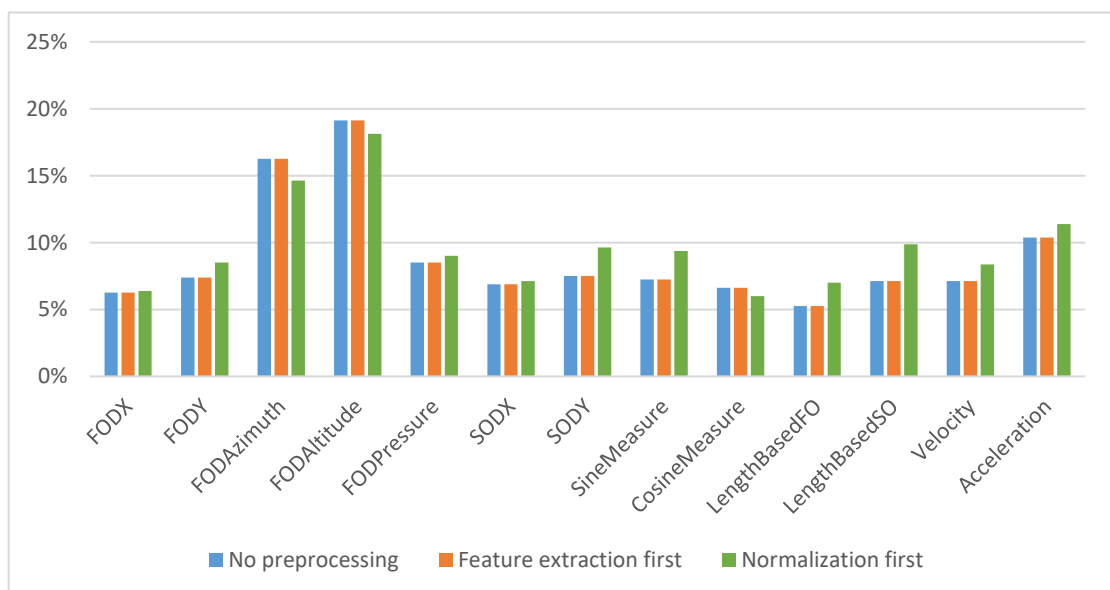
Az eredményeket különböző, beszédesen elnevezett .xlsx kiterjesztésű táblázatok formájában kapja a felhasználó, melyek aláíronként elért átlagos FAR, FRR és AER értékek, valamint a teljes adatbázison (40 aláíró) elért átlageredményeket. A részeredmények megjelenítésére az aláírások közt számított távolságok tételes listáját, valamint az optimális osztályozó által az egyes határérték mentén végzett tesztek eredményeit használja a rendszer.

5 Mérési eredmények

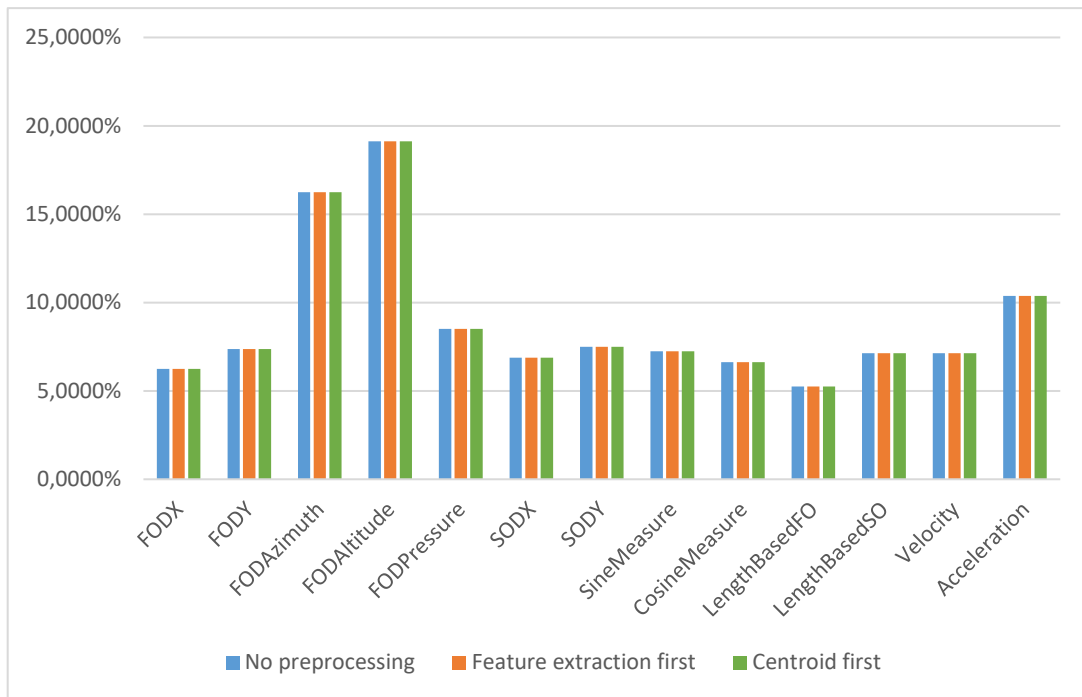
A korábbiakban bemutatott mérések mellett továbbiakat is végeztem az összefüggések felderítése céljából. Valamint a 3. fejezetben ismertetett különböző megközelítéseket a legjobb mérési eredmények alapján a legígéretesebb felállások szerint igyekeztem kombinálni, azzal a céllal, hogy további eredményjavulást idézzek elő.

Az előfeldolgozási lépések hatásának vizsgálatára végzett mérésekből kiderült, hogy abban az esetben, ha ezeket a már származtatott tulajdonságok értékein végezzük, akkor ezek semmilyen befolyással nem bírnak az eredményre. Természetes módon felvetődik a kérdés, hogy ezt biztosan a számított tulajdonságok jó jellemzőereje okozza-e, vagy az előfeldolgozó lépések késői használata. Ennek megválaszolása érdekében további méréseket végeztem az elméletileg elérhető legjobb AER értékeket vizsgálva, az alapmegközelítésben is szereplő DTW algoritmusra alapozva. Ezek a következő eredményeket hozták.

Amint azt a 18. ábra is mutatja normalizálás esetén akárhogy is választjuk meg a származtatott értékek kiszámításának és a normalizálási lépésnek a sorrendjét az nem eredményez pozitív változásokat. Sőt sok esetben a sorrend felcserélésével a normalizálás hozzáadása a rendszerhez még rosszabb eredményeket ad. Mindebből tehát arra következtethetünk, hogy alapvetően a származtatott értékek használata során a normalizációs lépésnek nincs gyakorlati haszna.



18. ábra: Tulajdonságszármaztatás és normalizálás egymáshoz viszonyított sorrendjének hatása a minimális AER elérését célzó, DTW alapú osztályozó eredményeire

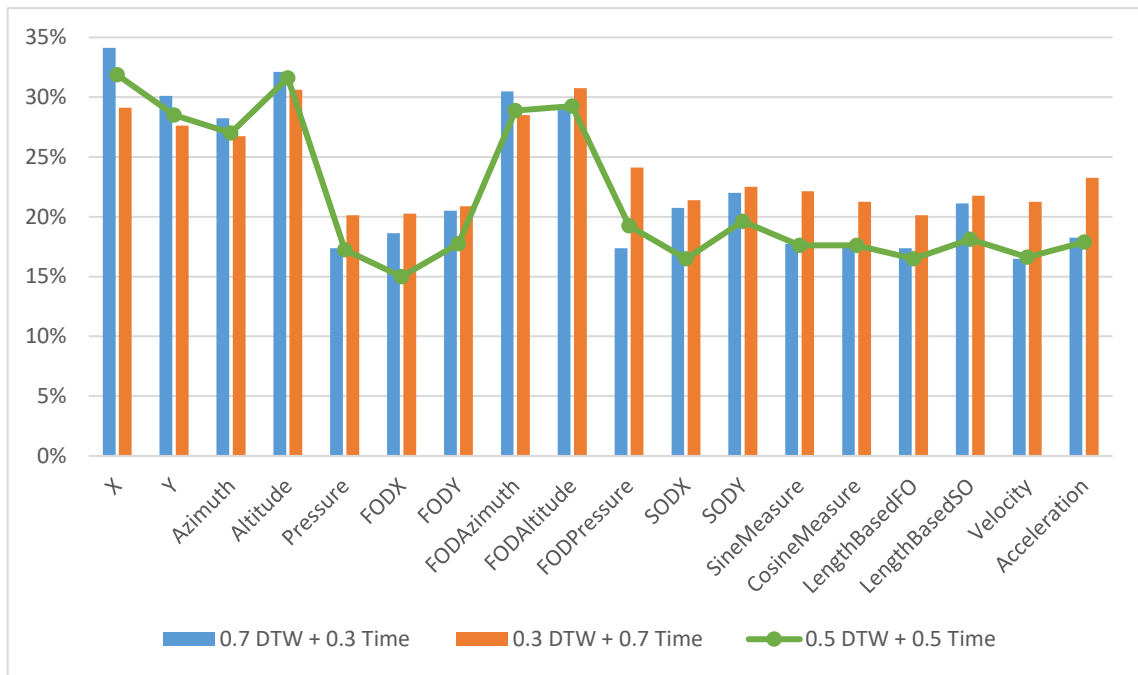


19. ábra: Tulajdonságszármasztás és súlypontba igazítás egymáshoz viszonyított sorrendjének hatása a minimális AER elérését célzó, DTW alapú osztályozó eredményeire

A 19. ábra egyértelműen mutatja, hogy a súlypontba igazítás szempontjából tökéletesen mindegy a sorrend, az semmiképp gyakorol hatást a szármasztott tulajdonságok vizsgálatával elérhető eredményekre.

Az alapvető méréseim alátámasztották, hogy a súlyozott, időbeli hossz alapján döntéseket hozó és valamilyen automatikus osztályozó összetételével kapott megoldások rendelkeznek létjogosultsággal. Ebből kifolyólag további méréseket végeztem annak vizsgálatára, hogy milyen hatással van az eredményekre, ha az összetett osztályozó tagjait nem azonos súllyal kezelem. Ezalatt kétféle súlybeállítással teszteltem az összetett osztályozót. Egyik esetben az idő alapú szűrést definiáltam megbízhatóbbnak az automatikus DTW osztályozóval szemben, ekkor a súlyok 0,7-0,3 voltak. Majd éppen fordítva, 0,7-0,3 súlyokkal a DTW osztályozó befolyását növeltem az idő alapú döntésekkel szemben.

A kapott eredményeket a 20. ábra tartalmazza, melyen látszik, hogy összességében az egyenlő súlyozással hozott döntések a legpontosabbak. Néhány esetben javítottak az eltoltsúlyozással definiált összetett osztályozók, ám a tulajdonságok nagy részénél rontottak az egyenrangú megközelítéssel ellentétben.



20. ábra: Automatikus DTW és időtartam alapú osztályozók kombinálása különböző súlyozások mellett

A korábbi mérések alapján meggyőződhetünk arról, hogy a tulajdonságpárok alkalmazásával pontosabb eredményeket kaphatunk az aláírások eredetiségéről hozott döntések során. A lehetőségeink azonban adottak ahhoz, hogy ne csak párokkal, hanem akár három vagy több tulajdonságból álló halmazokkal dolgozzunk. Ezen lehetőségek kiaknázása érdekében további mérésekre került sor, melyek során a DTW alapú osztályozó mind a két típusát teszteltem.

LengthBasedFO	FODX	CosineMeasure	SODX	Pressure	AER
x	x	x			3,375%
x	x		x		2,875%
x	x			x	3,375%
x		x	x		3,375%
x		x		x	5,375%
x			x	x	3,375%
	x	x	x		5,375%
	x	x		x	6,250%
	x		x	x	5,375%
		x	x	x	6,875%

7. táblázat: Minimális AER elérését célzó, DTW alapú osztályozó eredményeinek összehasonlítása az általa legjobban használható négy tulajdonságból képzett hármasok között

A 7. táblázat második sora bizonyítja számunkra, hogy megfelelően megválasztott három tulajdonság együttes vizsgálatával tovább lehet csökkenteni az elérhető AER alsó

határát. LenthBasedFO, FODX és SODX tulajdonságokat egyszerre vizsgálva a korábbi mérések során elért legjobb eredményen is sikerült 0,5%-ot faragni, és így 2,875%-os átlagos hibaarányt elérni.

FODPressure	Velocity	LengthBasedFO	SineMeasure	Pressure	AER
x	x	x			21,875%
x	x		x		21,250%
x	x			x	20,875%
x		x	x		22,000%
x		x		x	22,000%
x			x	x	24,750%
	x	x	x		21,875%
	x	x		x	21,875%
	x		x	x	21,000%
		x	x	x	22,000%

8. táblázat: Automatikus DTW alapú osztályozó eredményeinek összehasonlítása az általa legjobban használható négy tulajdonságból képzett hármasok között

Az automatikus osztályozó használata során ugyan a tulajdonságok hármas halmazával nem sikerült az eddigi legjobb eredményt felülmúlni, így is azonban átlagosan jobb eredményeket kaptunk, mint amikor tulajdonságpárokra futtattuk a DTW alapú automatikus osztályozót.

Ezután mind a két típusnál az általa legeredményesebben alkalmazott négy-négy tulajdonságra, mint halmazra vizsgáltam meg az eredményeket. A minimális AER elérését célzó osztályozó esetén a legjobb négy tulajdonság alapján és a nyomás hozzávételével kapott ötelemű halmaz alapján egyaránt az előzőekben elért 2,875%-os eredményt hozták a mérések.

Ezzel szemben az automatikus osztályozó meglepő módon a hármasok helyett a legjobb négy tulajdonságának mindegyikét figyelembe véve rontott a legeredményesebb hármashoz képest a maga 21,875%-os átlagos hibaarányával. A nyomás hozzávételével úgyszintén 21,875%-ot sikerült elérni.

Végül, de nem utolsó sorban az eddigiek alapján az automatikus DTW mellett legeredményesebbnek bizonyult tulajdonsághalmazokra az egyenlően súlyozott, összetett osztályozót futtattam. Ezáltal az automatikus DTW osztályozó legjobb eredményei születtek, melyeket a 9. táblázat foglalja össze.

FODPressure	Velocity	LengthBasedFO	SineMeasure	Pressure	AER
x	x	x	x		16,625%
x	x		x		14,750%
x	x			x	14,625%
	x		x	x	14,375%
x	x	x	x	x	16,500%

9. táblázat: Automatikus DTW alapú súlyozott osztályozó eredményeinek összehasonlítása legjobban használható tulajdonsághalmazok felett

Ezek az eredmények nem csak azért kiemelkedőek, mivel az eddigiek legjobbhöz képest is újabb 5,75%-os javulást hozta, hanem azért is, mert ezzel sikerült megközelíteni a minimális AER elérését célzó, DTW alapú osztályozó által az egyes tulajdonságokon elért 11%-os átlageredményét. Sikerült tehát egy olyan automatikus osztályozási konfigurációt definiálni, mellyel az alapmegközelítéssel elérhető elméleti minimum alá sikerült menni. Ne feledkezzünk meg róla, hogy az automatikus osztályozók jelen dolgozat keretei között kizárólag a referencia-aláírásokon mért távolságok átlaga és szórása alapján állítják be a határértéket. Ezt azt jelenti, hogy a határérték meghatározására használt módszer fejlesztésével lehetőségünk van olyan a gyakorlatban is használható automatikus osztályozót definiálni, amely az eredményeivel az elméleti határértékeket tovább közelítheti.

5.1 Eredmények értékelése

A kibővített mérésekkel sikerült elérni azoknak a célját, miszerint eredményeikkel alátámasztják a 3. fejezetben bemutatott eredményeket, összefüggéseket. Valamint szintén sikerrel jártak a különböző megközelítések megfelelő megválasztása mellett remélt eredményjavulás tekintetében.

Továbbá, ami szintén nagyon fontos, a szakirodalom által meghatározott eredményekkel nagyságrendileg összhangban vannak, azokkal összemérhetőek a jelen dolgozatban elért értékek.

6 Összegzés

Jelen dolgozatban különböző on-line aláírás-hitelesítési megközelítéseket vizsgáltam, hasonlítottam össze, és kombináltam. A megoldásaim tesztelésére, a mérések elvégzésére egy asztali alkalmazást fejlesztettem, mely teljes mértékben a kutatásom igényeit szolgálja ki.

Először bemutattam a tématerületet és annak jelentőségét. Ezután ismertettem a munkám szempontjából releváns szakirodalmat, a terület mai állását és a leggyakrabban használt megközelítéseket. Részletesen kitértem az általánosan alkalmazott eredményességi mutatókra, illetve az egyik legismertebb publikus adatbázisra, melyeket én is felhasználtam.

Ezután a saját munkám tételes leírása következett., Egyik első lépésként definiáltam az általam használt két különböző típusú osztályozó működését és célkitűzéseit. Továbbá ismertettem az általam alkalmazott megközelítéseket és azok alapvető mérési eredményeit. Majd röviden bemutattam a fejlesztett szoftvert is. Ezután néhány kibővített mérést mutattam be annak okán, célján és elért eredményein keresztül. Végezetül levontam azt a következtetést, hogy az általam elért eredmények az irodalommal összhangban vannak.

Mindeközben a minimális AER érték elérését célzó osztályozó használatával, az alapmegközelítés által elért 19,125%-os átlagos hibaarányt sikerült 2,875%-ra csökkentenem. Ehhez származtatott tulajdonságokat vezettem be, valamint megfelelően választottam meg a vizsgált tulajdonságok halmazát. Ez azt jelenti, hogy sikerült olyan módszert és hozzá konfigurációt találnom, amellyel az alapmegközelítéshez képest az elérhető legalacsonyabb elméleti AER értékét tekintve 16,25%-os javulást értem el.

Még nagyobb jelentőséggel bír, hogy az automatikus osztályozók esetén a kiindulási 36,375%-os AER értéket 14,375%-ra redukáltam. Ezzel a 22%-os javítással sikerült megközelíteni a DTW-vel egy tulajdonságot vizsgálva elérhető átlagos elméleti határértéket. Ez azért fontos, mert ezáltal megmutattam a megközelítések egy olyan együttesét, amivel egyszerű statisztikai mutatók alapján beállított határértékkel is már elég jó eredményt lehet elérni, és az osztályozó működése szempontjából a gyakorlatban is használható. A határérték számításának más módjával lehetőség nyílhat egy gyakorlati szempontból jelentős, eredményeit tekintve is jól hasznosítható megalkotására.

7 Irodalomjegyzék

- [1] T.Sabhanayagam, D. V. P. Venkatesan és D. K. Senthamaraikannan, „A Comprehensive Survey on Various Biometric Systems,” *International Journal of Applied Engineering Research*, %1. kötet13, %1. szám5, pp. 2276-2297, 2018.
- [2] F. D. G. S. D. C. Anil K. Jain, „On-line signature verification,” *Pattern Recognition*, %1. kötet35, %1. szám12, pp. 2963-2972, 2002.
- [3] H. Y. Kai Huang, „Off-line signature verification based on geometric feature extraction and neural network classification,” *Pattern Recognition*, %1. kötet30, %1. szám1, pp. 9-17, 1997.
- [4] H. P. D.S. Guru, „Online Signature Verification and Recognition: An Approach Based on Symbolic Representation,” *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, %1. kötet31, %1. szám6, pp. 1059-1073, 2009.
- [5] J. M. P.-G. C.-P. E. Vivaracho-Pascual, „Practical On-Line Signature Verification,” in *Advances in Biometrics*, Springer Berlin Heidelberg, 2009, pp. 1180-1189.
- [6] A. Sharma és S. Sundaram, „A Novel Online Signature Verification System Based on GMM Features in a DTW Framework,” *IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY*, %1. kötet12, %1. szám3, pp. 705-718, 2017.
- [7] D. Impedovo és G. Pirlo, „Automatic Signature Verification: The State of the Art,” *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C: APPLICATIONS AND REVIEWS*, %1. kötet38, %1. szám5, pp. 609-635, 2008.
- [8] Xinghua Xia, Zhili Chen, Fangjun Luan és Xiaoyu Song, „Signature alignment based on GMM for on-line signature verification,” *Pattern Recognition*, %1. kötet65, pp. 188-196, 2017.

- [9] M. T. Ibrahim, M. A. Khan, K. S. Alimgeer, M. K. Khan, I. A. Taj és L. Guan, „Velocity and pressure-based partitions of horizontal and vertical trajectories for on-line signature verification,” *Pattern Recognition*, %1. kötet43, pp. 2817-2832, 2010.
- [10] Upasna Jindal, Surjeet Dalal és Neeraj Dahiya, „A combine approach of preprocessing in integrated signature verification (ISV),” *International Journal of Engineering & Technology*, pp. 155-159, 2018.
- [11] Xiaoyu Song, Xinghua Xia és Fangjun Luan, „Online Signature Verification Based on Stable Features Extracted Dynamically,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, %1. kötet47, %1. szám10, pp. 2663 - 2676, 2017.
- [12] Alisher Kholmatov és Berrin Yanikoglu, „Identity authentication using improved online signature verification method,” *Pattern Recognition Letters*, %1. kötet26, %1. szám15, pp. 2400-2408, 2005.
- [13] M. Faundez-Zanuy, „On-line signature recognition based on VQ-DTW,” *Pattern Recognition*, %1. kötet40, %1. szám3, pp. 981-992, 2007.
- [14] H. Feng és C. C. Wah, „Online signature verification using a new extreme points warping technique,” *Pattern Recognition Letters*, %1. kötet24, %1. szám16, pp. 2943-2951, 2003.
- [15] A. Parziale, M. Diaz, M. A. Ferrer és A. Marcelli, „SM-DTW: Stability Modulated Dynamic Time Warping for signature verification,” *Pattern Recognition Letters*, pp. 1-10, 2018.
- [16] Loris Nanni, „Experimental comparison of one-class classifiers for online signature verification,” *Neurocomputing*, %1. kötet69, %1. szám7-9, pp. 869-873, 2006.
- [17] P. Senin, „Dynamic Time Warping Algorithm Review,” Honolulu USA, December 2008.
- [18] Yu Qiao, Xingxing Wang és Chunjing Xu, „Learning Mahalanobis Distance for DTW based Online Signature Verification,” in *IEEE*, Shenzhen, China, 2011.
- [19] N. Sae-Bae és N. Memon, „Online Signature Verification on Mobile Devices,” *IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY*, %1. kötet9, %1. szám6, pp. 933-947, 2014.

- [20] M. I. Malik, S. Ahmed, A. Marcelli, U. Pal, M. Blumenstein, L. Alewijns és M. Liwicki, „ICDAR2015 competition on signature verification and writer identification for on- and off-line skilled forgeries (SigWlcomp2015),” in *IEEE*, Tunis, Tunisia, 2015.
- [21] M. Liwicki, M. I. Malik, C. E. v. d. Heuvel, X. Chen, C. Berger, R. Stoel, M. Blumenstein és B. Found, „Signature Verification Competition for Online and Offline Skilled Forgeries (SigComp2011),” in *IEEE*, Beijing, China, 2011.
- [22] M. I. Malik, M. Liwicki, L. Alewijns, W. Ohyama, M. Blumenstein és B. Found, „ICDAR 2013 Competitions on Signature Verification and Writer Identification for On- and Offline Skilled Forgeries (SigWiComp 2013),” in *IEEE*, Washington, DC, USA, 2013.
- [23] D.-Y. Yeung, H. Chang, Y. Xiong, S. George, R. Kashi, T. Matsumoto és G. Rigoll, „SVC2004: First International Signature Verification Competition,” in *Biometric Authentication*, Berlin, Heidelberg, Springer Berlin Heidelberg, 2004, pp. 16-22.
- [24] A. Sharma és S. Sundaram, „On the Exploration of Information From the DTW Cost Matrix for Online Signature Verification,” *IEEE TRANSACTIONS ON CYBERNETICS*, %1. kötet48, %1. szám2, pp. 611-624, 2018.
- [25] E. Griechisch, „ONLINE SIGNATURE VERIFICATION AND HANDWRITING CLASSIFICATION,” 2018.
- [26] A. Sharma és S. Sundaram, „An enhanced contextual DTW based system for online signature verification using Vector Quantization,” *Pattern Recognition Letters*, %1. kötet84, pp. 22-28, 2016.
- [27] „Signature verification library,” [Online]. Available: <https://github.com/sigstat/sigstat>.