



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

**ELŐFELDOLGOZÁSI LÉPÉSEK
HATÁSA A DTW ALAPÚ
ONLINE ALÁÍRÁSHITELESÍTÉSI
ALGORITMUSOK
PONTOSSÁGÁRA**

KONZULENS

Dr. Kővári Bence

BUDAPEST, 2019

Tartalomjegyzék

Összefoglaló	4
Abstract.....	5
1 Bevezetés	6
1.1 Online aláíráshitelesítés	7
1.2 Online aláíráshitelesítés jelentősége	8
2 Irodalmi áttekintés.....	10
2.1 Előfeldolgozás	10
2.2 Hitelesítési szempontok	11
2.3 Aláírások összehasonlítására használt módszerek	12
2.3.1 Dynamic Time Warping	12
2.3.2 További algoritmusok	13
2.3.3 Távolságmértékek	15
2.4 Eredmények értékelése	15
2.5 Adatbázisok	16
3 Korábbi kutatási tevékenységem.....	18
3.1 Osztályozók típusai	18
3.1.1 Minimális AER elérését célzó osztályozó	19
3.1.2 Automatikus osztályozó.....	20
3.2 Aláírások összehasonlítására alkalmazott algoritmus.....	20
3.2.1 DTW algoritmus megvalósítása.....	20
3.3 Aláírástulajdonságok megválasztása	22
3.3.1 Tulajdonságok kombinációi.....	22
3.4 Előfeldolgozás	22
4 Előfeldolgozási algoritmusok	24
4.1 Újramintavételezési algoritmusok	24
4.1.1 Interpoláció	24
4.1.2 Újramintavételezés a pontok fix száma alapján.....	26
4.1.3 Nulla nyomású pontok eltávolítása.....	27
4.1.4 Tollfelemelések kitöltése	27
4.2 Elforgatás	28
4.3 Eltolás	30

4.4 Skálázás	33
5 Egyéb hitelesítési paraméterek	35
6 Mérési keretrendszer	36
6.1 Elvégzett mérések	36
7 Mérési eredmények.....	38
7.1 Pontok fix száma alapján végzett újramintavételezés hatása.....	38
7.2 Elforgatás hatása	39
7.3 Távfolságfüggvény megválasztása.....	41
7.4 Eltolás hatásai	42
7.5 Skálázás hatásai	45
7.6 Eredmények értékelése	47
8 Összegzés.....	48
9 Irodalomjegyzék.....	49

Összefoglaló

A kézi aláírás az egyik legrégebb óta és legszélesebb körben használt biometrikus hitelesítési eszköz. Napi rendszerességgel írunk alá különböző hivatalos dokumentumokat, ám a technológiai változásokat követve egyre többször találkozunk digitális táblagépek, aláírópadok használatával a megszokott papír alapú nyomtatványok helyett. A régen megszületett elvárás, hogy egy-egy aláírás eredetiségéről pontos döntést tudjunk hozni, valamint a technológiai fejlődés által biztosított új lehetőségek közösen hívták életre az automatizált online aláírás-hitelesítés igényét.

Ennek során a különböző digitális segédeszközökkel (pl. nyomásérzékeny toll, digitális táblagépek stb.) rögzített aláírások eredetiségéről születik korlátos időn belül egy programozott döntés. A cél, hogy ezeket a döntéseket a hitelesítő rendszer minél nagyobb biztonsággal, a lehető legkevesebb tévedéssel tudja meghozni. Az aláírás-hitelesítő rendszerek ezért nagy bonyolultságú, összetett megoldások, melyek különböző fázisokban, rengeteg szempont szerint vizsgálják az aláírásokat.

Jelen dolgozat célja a Dynamic Time Warping (DTW) algoritmus használatán alapuló hitelesítő rendszerek különböző lehetséges elemeinek (hitelesítési szempontok, algoritmusok), azok megválasztásának a hitelesítési eredményekre gyakorolt hatásának vizsgálata. Különös tekintettel az aláírások rögzítésekor keletkezett egyes torzulások, anomáliák megszüntetését célzó, úgynevezett előfeldolgozó lépésekre. A dolgozatban szereplő megfigyelések egy széleskörű mérésorozat eredményeit képezik, így általánosabbak, nem csupán egy-egy speciális esetben helytállóak. A dolgozat célja továbbá a mérések során tapasztalt megfigyelések, megállapítások segítségével elősegíteni és támogatni hatékony online aláírás-hitelesítő rendszerek tervezését.

Régebb óta foglalkozom a DTW alapú online aláírás-hitelesítés témakörével, ennek köszönhetően jelen dolgozatom számos ponton épít a 2018. őszén beadott „Online aláírás-hitelesítési algoritmusok vizsgálata” című TDK dolgozatomra. Az elmúlt egy évben végzett kutatás a korábbiakon alapszik, ám a tavalyi dolgozat sokkal szűkebb metszete volt a témakörnek, jelen dolgozat több adatbázison végzett méréseket dokumentál, jóval több előfeldolgozó algoritmust vizsgál, sokkal részletesebben, valamint vizsgálja a referenciaként használt aláírások korának hatásait is.

Abstract

Handwritten signature is one of the biometrics, which has been widely applied for centuries. Handwritten signatures are used almost every day, but because of the new technologies the usage of special electronic devices (e.g. tablets) is more and more common instead of the traditional printed papers. The ability to decide about the origin of unknown signatures is an old expectation, but the technological changes enable the opportunity of automatic online signature verification.

During this, signatures are acquired with the help of different electronic devices (e.g. pressure sensitive pen, digital tablets etc.), and the goal is to decide about these captured signatures whether they are genuine or forgeries. The decision must be made in polynomial time and as precisely as possible. Thus, signature verification systems are very complex, they contain several phases and apply various number of parameters during the verification.

The goal of this paper is to analyse the effects of the usage of different elements, such as verification parameters and algorithms, on the accuracy of Dynamic Time Warping (DTW) based online signature verification system. There may be some distortion and noise in the captured signature data, so this paper pays special attention to the effects of different preprocessing approaches, which try to eliminate these mentioned problems. This paper presents the results of comprehensive experiments. The goal of this paper is to help the construction of effective signature verification systems as well.

I have been researching DTW based online signature verification for a long time, thus this paper refers to my previous paper named „On-line aláírás-hitelesítési algoritmusok vizsgálatá” (Study of online signature verification algorithms) from 2018. My research from the last year is based on my previous work as well, but the previous paper presented only a much smaller slice of this field. This work was evaluated on several databases, it contains more preprocessing approaches and they are presented in more details. This paper deals with the importance of the age of the reference signatures as well.

1 Bevezetés

A kézzel írott aláírások rengeteg egyedi jellemzőt hordoznak magukban, melyeknek egyértelműen az adott emberhez köthetők. Ennek is köszönhetően már nagyon régóta használjuk azokat különböző hivatalos dokumentumok ellenjegyzésére, önmagunk írásban történő azonosítására. Az aláírás mai napig az egyik leggyakrabban használt, széleskörben elfogadott biometrikus hitelesítési eszköz [1].

Az aláírások használatának jelentősége miatt fontos elvárás a hamisítványok kiszűrése, ezzel megakadályozva az esetleges visszaéléseket. Szerencsére az egyes aláírók egyedi ismertetőjegyei biztosítják is a feltételeket ennek kivitelezéséhez. Megfelelő szakemberek (írasszakértők) nagy pontossággal képesek eldönteni egy adott aláírásról, hogy eredeti-e, természetesen ehhez szükségük van eredeti aláírásmintára is. Az írasszakértők száma azonban korlátozott, és nem is minden esetben van lehetőség a jelenlétükre. Mint oly sok minden másra az emberiség fejlődése során erre a problémára is megoldásként szolgálhat a technológia, így napjaink egy fontos törekvése az aláíráshitelesítés automatizálása. Az automatizált aláíráshitelesítés lehetővé teszi, hogy korlátos időn belül, szoftveres döntés születhessen az ismeretlen eredetű aláírások eredetiségéről.

A pontos döntések érdekében gyakran szükség van az adatok előkészítésére, ami során a hitelesítés pontosságát torzító tényezők kiküszöbölése a cél. Szinte lehetetlen például akár csak kétszer egymás után ugyanúgy aláírni, hogy a papír pontosan ugyanazon pontján kezdődjön az aláírás és azonos pontban is érjen véget, pontosan egyező méretű legyen és a vízszinteshez képest is ugyanolyan arányban legyen elforgatva. Ezek a tényezők az eredeti aláírások közti különbségeket indokolatlanul növelhetik, ezzel tévesen befolyásolva a hitelesítési döntést. Szerencsére azonban különböző előfeldolgozási lépések alkalmazásával van lehetőség a félrevezető különbségek megszüntetésére és az aláírások lecsupaszítására ebből a szempontból. Az aláírásokat előzetesen egységesen valamilyen referenciapontba tolva, azonos méretre skálázva és vízszintesbe forgatva lehetővé tesszük, hogy a hitelesítő rendszernek csak az érdemi eltéréseket kelljen figyelembe vennie. A lokalizációs problémákon túl, persze technikai tényezők is kerülhet további zaj az aláírásadatokba. Ez szintén torzíthatja az eredményeket, de az előfeldolgozó lépések erre is megoldást jelenthetnek. Jelen

dolgozatom fókuszában az ilyen előfeldolgozó lépések állnak. Egy-egy megközelítés több konkrét algoritmussal is megvalósítható, én ezekből implementáltam jónéhányat, majd egy átfogó mérésorozat eredményei alapján ezeket hasonlítottam tételesen össze. Mindezzel a célom az volt, hogy segítsen az egyes megközelítések, és azon belül a konkrét megvalósítások közti eredményes választást.

Az adatok megtisztítása után következik maga a hitelesítés feladata. Ez egy két osztályos osztályozási probléma, mely során egy adott aláírást az eredeti vagy a hamis osztályba kell besorolni. A döntéshez minden esetben szükség van eredeti referencia-aláírásokra, amelyekkel az ismeretlen eredetű aláírást össze lehet vetni, és a hozzájuk viszonyított hasonlósága alapján lehet meghozni a döntést. Természetesen az emberek általában nem írnak kétszer sem ugyanúgy alá, így a feladat nehézsége megtalálni azt a határt, ami az eredeti és hamis aláírásokat elválasztja egymástól.

1.1 Online aláíráshitelesítés

Az online aláíráshitelesítés az automatizált aláíráshitelesítés egy válfaja, melynek lényege továbbra is az aláírások szoftveres vizsgálata és az eredetiség kérdésének ilyen módon történő eldöntése. A terület újdonságát az aláírások tárolásának formátuma adja. Jelen dolgozatom is online aláíráshitelesítéssel foglalkozik.

A folyamatos technológia fejlődésnek hála, a mindennapjainkat is behálózó digitalizáció miatt egyre több helyen találkozhatunk papírmentes működéssel, hogy a szerződések, leletek és különféle dokumentációk tárolása már elsődlegesen csak elektronikusan történik. Ennek megfelelően egyre ritkábban találkozunk hagyományos tollal papírra vetett aláírásokkal, helyette egyre többször használunk digitális aláírópadot vagy egyszerű digitális táblagépeket.

Az online aláíráshitelesítés során nem az aláírások képét tároljuk és használjuk fel, hanem az egyes aláírásokat, mint egy-egy adathalmazt lehet leírni. A fent említett különböző digitális eszközök lehetővé teszik továbbá az aláírás kétdimenziós képe által is tárolt koordinátainformációk mellett további dinamikus tulajdonságok rögzítését is. Ez azért nagyon fontos, mert ezeket a dinamikus jegyeket még nehezebb hamisítani, hisz nem csak az aláírás végeredményét, hanem magát a folyamatot és annak dinamikáját is leírják.

1.2 Online aláíráshitelesítés jelentősége

Az aláírás használata a hétköznapiak során is nagyon gyakori és sokszor igen nagy jelentőséggel bír, például nagy értékű szerződések megkötésénél. Emiatt bár alkalmazásának technikai megoldásai az idővel változnak, meghatározó szerepe ez idáig mégis változatlan. Ezt tükrözi az is, hogy az utóbbi évtizedekben ez egy aktívan kutatott terület lett. Sokan foglalkoznak a terület problémáinak megoldásával, melynek eredményeképp számos különböző megközelítés és algoritmus született.

A tudományos munka és az ennek során megalkotott egyes aláíráshitelesítő rendszerek bemutatására a hagyományos publikációkon túl, úgynevezett aláíráshitelesítő versenyek is lehetőséget biztosítanak. Az első ilyen megmérettetés az SVC2004: First International Signature Verification Competition [2] volt, amit azóta is számos másik követett. Az ICDAR konferenciasorozat is rendelkezik aláíráshitelesítő versennyel, mely a Competition on Signature Verification and Writer Identification for On- and Off-line Skilled Forgeries (SigWiComp) nevet viseli. Ez a megmérettetés 2009. óta két évente, összesen négy alkalommal (2009. [3], 2011. [4], 2013. [5], 2015. [6]) került eddig megrendezésre.

Ezeknek a versenyeknek az aktuális élvonalbeli megoldások bemutatásán és összehasonlításán túl, további hosszú távú hozadékuk is lett, ugyanis tovább gazdagították a nyilvánosan elérhető aláíradatbázisok halmazát. A megmérettetések döntő többségében nem csak online, de offline a dinamikus információktól mentes aláírásképeket tartalmazó adatbázisokat is publikáltak, melyeken a nevezett rendszereket tanítani és tesztelni tudták a résztvevők. Számos adatbázis azonban ennél jóval hosszabb életűnek bizonyult, és a mai napig gyakran használják őket az irodalomban egy-egy megoldás eredményességének tesztelésére. Ennek előnye, hogy az ugyanazon az adatbázison elért eredmények alapján jobban össze lehet hasonlítani különböző rendszereket, algoritmusokat és megközelítéseket.

A tudományos kutatásokon felül mára az iparban is találkozhatunk gyakorlati alkalmazásokkal is, ugyan ezekről érthető módon nem érhető el túl sok információ. Mindenesetre felhasználóként vagy ügyfélként is mindannyian tapasztalhatjuk a postán, bankfiókokban, valamilyen telekommunikációs szolgáltató ügyfélszolgálatán vagy épp a futárral kézbesített csomagok átvételkor, hogy egyre többször kell az ujjunkkal vagy speciális tollal aláírunk különböző érintőképernyős eszközökön.

Az online aláírások hitelesítés kérdésének megoldása egy nagy megbízhatóságú rendszerrel a meglévő alkalmazási területeken kívül akár új lehetőségek előtt is kaput nyithat. Amennyiben az aláírás azonnal digitális formában is elérhető és megvannak a megfelelő biztonságos hitelesítési eszközeink hozzá, akkor akár az autentikáció során is rengeteg alkalmazásban kaphatna teret. Az online aláírást érezhetően nehezebb elloponi, kitalálni vagy hamisítani, így akár felválthatnák a pin kódok és jelszavak használatát [7].

2 Irodalmi áttekintés

Ebben a fejezetben a tudomány jelenlegi állását, az ezidáig használt és kutatott aláíráshitelesítési megoldásokat szeretném nagy vonalakban ismertetni. Mindezt különös tekintettel a munkám szempontjából releváns DTW alapú megközelítésekre, valamint a jelenlegi kutatásom fókuszában álló különböző előfeldolgozási algoritmusokra. A szakirodalom bemutatására a méréseim konfigurációja során megadható egyes paraméterek mentén kerül sor.

2.1 Előfeldolgozás

Amint azt a korábbiakban is írtam az aláírások rögzítésekor számos olyan tényező játszik szerepet, mely torzíthatja az aláíras adatokat, és ezáltal az alkalmazott algoritmustól függő mértékben az eredményekre is befolyással bírhatnak. A különböző eszközök minőségük, vagy beállításuk következtében további zajt is rögzíthetnek, ami szintén jelentős hatást gyakorolhat a hitelesség pontosságára. Ezek negatív hatásának megszüntetésére használunk előfeldolgozást azzal a céllal, hogy ezzel csökkentjük az eredeti és hamis aláírások osztályon belüli különbségeit.

Az aláírás pozicionálásából fakadó különbségek megszüntetése céljából gyakran alkalmaznak eltolást valamilyen formában [8] [9] [10]. Sok esetben az aláírások méretében is van különbség, ezért az ebből adódó torzító hatás kiküszöbölésére sűrűn alkalmazott megoldás a skálázás [9] [11] [12] [10] [13]. Az ugyancsak jellemző elfordulását az aláírásoknak szintén lehet az előfeldolgozás során orvosolni, egészen egyszerűen a megfelelő szöggel el kell forgatni az aláírást [13] [8] [12] [10].

Az aláírás begyűjtése során rögzített zaj is kiszűrése szintén nagyon fontos és gyakran alkalmazott adatelőkészítő lépés [14]. Érdekes kérdés továbbá, hogy hogyan kezeljük azokat a pontokat, amik rögzítésekor a toll éppen a levegőben volt. Vannak olyan eszközök, melyek képesek a Z-tengely menti távolságot mérni, ám amennyiben a hitelesítés során ezt a szempontot nem vizsgáljuk, akkor a többi tulajdonság tekintetében ezen pontok információtartalma változhat. Az egyik lehetőség ezeknek a nulla nyomású pontoknak az eltávolítása [9], míg a másik lehetőség ezeknek a megtartása. Utóbbi esetben felmerül a kérdés, hogy az egységes kezelés érdekében ezt hogyan lehet megoldani olyan aláírásokon, amelyek valamilyen okból kifolyólag kizárólag a nem nulla

nyomású pontok adatait tartalmazzák. Ebben az esetben valamilyen interpolációs megközelítés alkalmazásával lehetőség nyílik ezeknek a pontoknak a generálására.

Az előkészítő lépések szerepe nem merül ki csupán a hitelesítési eredmények javításában, ezek ugyanis a hitelesítő rendszerek, mint szoftverek teljesítményét, hatékonyságát, gyorsaságát is növelhetik.

2.2 Hitelesítési szempontok

Ahogy az korábban is említettem, az online aláíráshitelesítés során használt digitális eszközök számos dinamikus tulajdonságot is képesek rögzíteni egy-egy aláírásról, nem csupán a koordinátainformációkat. Ez a tény számtalan új lehetőséget biztosít az aláírások hitelesítésének alapját képező szempontrendszer megválasztására, így ennek helyes megválasztása még fontosabb feladat.

Az online aláírások négy legalapvetőbb tulajdonsága az X- és Y-koordináták, a nyomás és az időbélyeg. Ezek közvetlenül az aláírások rögzítésekor már rendelkezésre állnak, és igen gyakran alkalmazzák különböző online hitelesítő rendszerek.

Gyakran rögzítik a toll Z-tengely menti elmozdulását is, amit olykor a papírtól vett távolságának számszerű értékeként, máskor binárisan tárolnak. Utóbbi esetben lényegében csak azt különböztetik meg, hogy a toll az adott időpillanatban éri-e a papírt vagy a levegőben van. Ez valójában a nyomás értékekből is kiolvasható lenne, hiszen csupán a nulla és nem nulla nyomású pontokat kell megkülönböztetni. Ezen kívül az aláírás egyes pontjaiban meghatározott két szögmértéket is sűrűn alkalmaznak, ezeket azonban nem minden publikusan hozzáférhető online aláírástárház adatainak felvételekor rögzítették. Az egyik ilyen tulajdonság az úgynevezett irányszög vagy azimuth, mely a toll az óramutató járásával megegyező irányú, Z-tengely körüli elfordulását határozza meg. A toll és a pozitív Z-tengely által felfelé bezárt szöget az irodalomban altitude, azaz magasság néven szokták említeni.

Mivel a dolgozatomban célja az előfeldolgozó lépések összehasonlítása, így a mérések során kizárólag az ilyen alaptulajdonságokat alkalmaztam.

Az osztályozás során az aláírások jellemzésére használt tulajdonságok megválasztásának lehetőségeit tovább gyarapíthatjuk a rögzítő eszközök továbbfejlesztése nélkül, a rögzített alaptulajdonságokból származtatott új tulajdonságok meghatározásával is. Új tulajdonságokat a legkülönfélébb ötletek szerint definiálhatunk.

Legkézenfekvőbb lehet például az alaptulajdonságok deriváltjainak használata [15] vagy valahányad rendű különbségek, távolság alapú egyéb mértékek, esetleg különböző szögmértékek alkalmazása [16]. További már alkalmazott lehetőség például intervallumreprezentánsok bevezetése [17], de a lehetőségek tárháza még ennél is sokkal széleskörűbb [18].

Az új levezetési módszerek definiálása nem az egyetlen lehetőség az eredmények javítására, ugyanis a legtöbb osztályozási megoldás képes tulajdonságok egy tetszőleges halmazát felhasználni az összehasonlítások során, nem csak pusztán egy-egy aláírástulajdonságot. A megfelelő kombinációk meghatározására szintén számos lehetőség van, ami nagyban befolyásolhatja a hitelesítési eredményeket [15].

Mindezekből jól látszik, hogy bár vannak gyakrabban alkalmazott tulajdonságok, ennek ellenére nincs egy olyan fix tulajdonsághalmaz definiálva minden esetben alkalmazandó és jól is alkalmazható. Az eredmények javítása céljából fontos feladat, hogy a hitelesítési szempontok megválasztása összhangban legyen a hitelesítési algoritmus lehetőségeivel.

2.3 Aláírások összehasonlítására használt módszerek

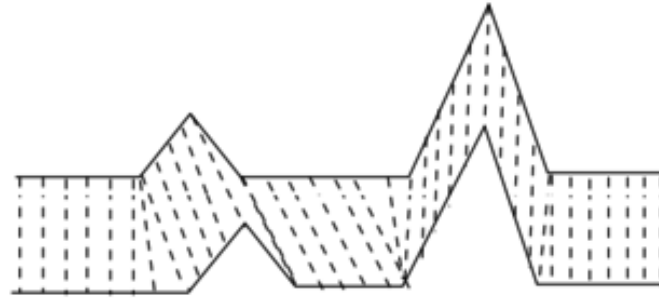
A másik nagyon meghatározó lépése az aláíráshitelesítő rendszer tervezésének az aláírások összehasonlítására használt algoritmusok megválasztása. Ezekre is számtalan különböző megközelítés született már a pontosabb eredmények reményében, a célja azonban mindegyiknek közös. Két aláírás közt fennálló távolságot valahogyan úgy számszerűsíteni, hogy az eredeti aláírások közt lehetőleg kis, míg eredeti és hamis közt érzékelhetően nagyobb értéket kapjunk.

A következőkben először szeretném részletesebben is bemutatni az általam is használt Dynamic Time Warping (DTW) algoritmust, majd röviden további ennek használatán alapuló megoldásokat.

2.3.1 Dynamic Time Warping

A DTW algoritmus [19] sok aláíráshitelesítő rendszernek szolgál alapjául, ezen a területen nagyon széles körben elterjedt. Elsőként a beszédfelismerésben alkalmazták a DTW algoritmust, mára azonban nem csak az aláíráshitelesítés, hanem számtalan további terület átvette a használatát. Ilyen például az adatbányászat és idősorok klaszterezése stb.

A DTW algoritmus, hogy két idősor közötti hasonlóságot meghatározza. Mindezt nagyon hatékonyan képes megtenni, ráadásul az időbeli elcsúszást és torzulást is minimalizálni tudja, ezért nagy népszerűségnek örvend. Az algoritmus bemenete két idősor reprezentáló értéksorozat. Ezek 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.



1. ábra: DTW algoritmus alapötletének sematikus rajza

A hasonlóság meghatározására a sorozatok közti távolságfüggvényt használjuk. Minél kisebb a távolság a két sorozat között, annál hasonlóbbak. Azonos logika mentén minél nagyobb ez a távolság, annál jobban különböznek egymástól. A távolságfüggvényt más néven költségfüggvénynek is szokták nevezni.

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 megvalósításról a DTW algoritmus megvalósítása című fejezetben olvasható.

2.3.2 További algoritmusok

Léteznek olyan megoldások is, amik tisztán a DTW algoritmust alkalmazzák, de természetesen a DTW algoritmusnak is vannak korlátjai, lehetnek gyengeségei. Számptalan továbbfejlesztési megoldás létezik ezért, melyek célja az eredeti DTW hatékonyságának növelése.

Egy megközelítés lehet erre, ha a DTW által meghatározott értékeket használjuk fel, ám azok mentén az eredeti és hamis aláírások közti határvonal meghúzásához valamilyen egyéb módszert alkalmazunk. Ekkor a tanító fázisban, a referencia-aláírások vizsgálata során szeretnénk elsősorban pontosabb eredmények elérését biztosítani. Ilyen

megközelítés például a support vector machine (SVM) vagy Bayes osztályozó bevonása. Ezek alkalmazása mellett 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. [20]

Egy másik megközelítés lehet az eredeti DTW kombinálása valamilyen más módszerrel. A [21] cikkben például a vektor kvantálás (vector quantization VQ) módszerével ötvözték. A kombinációs megoldások mellett magának a DTW algoritmus módosítására, továbbfejlesztésre is lehetőség van, mely célja szintén a DTW hiányosságainak, gyengeségeinek kiküszöbölése. Így született új technika például az extreme point warping (EPW) [22], mellyel a dinamikus tulajdonságok bevonása nélkül, DTW-vel elért 33%-os EER (Equal Error Rate) értéket EPW-vel 25,4%-ra sikerült csökkenteni. Szintén egyfajta továbbfejlesztése az SM-DTW, vagyis Stability Modulated Dynamic Time Warping [23], amivel az MCYT-100 elnevezésű adatbázison 3,09%-os EER-t sikerült elérni profi hamisítványok mellett. Okawa cikkében [24] a DTW két módosított változatát is bemutatja, mely a több tulajdonság alapján történő hitelesítés során alkalmazható. A két megközelítést az különbözteti meg, hogy a különböző tulajdonságokat egymástól függetlenül vagy függően kezeli a távolság megállapítása során. A cikkben publikált rendszerével, mely a függőségi megoldásra épít 1,34%-os EER-t tudott elérni az MCYT-100 adatbázison. A [25] cikkben bemutatott shape context-dynamic time warping (SC-DTW) néven bemutatott módosított DTW algoritmussal az SVC2004 Task2 adatbázison 2,39%-os EER értéket sikerült elérni.

A DTW algoritmus elterjedtsége mellett természetesen egészen más megoldások is léteznek. Ilyenek 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. [26] Ezen felül létezik megoldás, mely fuzzy similarity measure alkalmazásával működik [27]. Továbbá számos megvalósítás van, mely a más területeken is népszerűségnek örvendő neurális hálók használatán alapszik [28] [29] [30].

2.3.3 Távolságmértékek

Sok módszer, ahogy a DTW algoritmus is, végeredményben valamilyen klasszikus távolságmértéken alapszik, azt használja működéséhez. Ennek megválasztása szintén az eredmények egy befolyásoló tényezője lehet. Az egyik leggyakrabban alkalmazott ilyen például az egyszerű Euklideszi-távolság. A Manhattan-távolság használata, melyet city block distance néven is gyakran említenek szintén sűrűn előfordul. Ebben a dolgozatban az előbbi két távolságdefiníció közti eredményes választást is vizsgálom, használatuk hitelesítési eredményekre gyakorolt hatásának szempontjából. A klasszikusabb megközelítések mellett persze használnak más megoldásokat is, például a Mahalanobis-távolságot. [31]

2.4 Eredmények értékelése

A különböző aláíráshitelesítési megoldások helyes működését és jó eredményeit valahogyan ellenőrizni szeretnénk. Az alapvető cél minden rendszer esetében az, hogy helyes döntéseket hozzon, vagyis az eredeti aláírásokat eredetinek, míg a hamisakat hamisnak ítélje. A különböző módszerek pontosságának meghatározására jellemzően négyféle hibaarányt szoktak használni, melyek egységes használata lehetővé teszi az irodalomban fellelhető különböző megoldások összehasonlíthatóságát. Értelemszerűen minél kisebb értéket vesznek fel ezek a mutatók, annál pontosabb és megbízhatóbb az adott hitelesítési megközelítés eredménye. Az alábbiakban ezt a négy hibaarányt szeretném ismertetni.

Alapvetően kétféle hibás döntés lehetséges az aláíráshitelesítés során. Az első esetben egy eredeti aláírás kerül tévesen elutasításra, erre a hibatípusra az angol terminológia szerint Type I Error néven hivatkoznak. Az ilyen módon bekövetkezett hibák arányát mutatja meg a False Rejection Rate (FRR), melynek jelölése α , kiszámítási módja pedig a következő képlet segítségével írható le:

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

A másik lehetőség, vagyis a Type II Error az az eset, amikor a rendszer tévesen elfogad eredetiként egy hamisítványt. Az aláírás-hitelesítés során bekövetkezett ilyen típusú hibák arányát False Acceptance Rate néven említik legtöbbször, jelölése β . Kiszámítási módja a következő képlet segítségével írható le:

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

A fenti két hibaarány 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 minimalizálása, vagyis FRR és FAR lehető legkisebb értékét szeretnénk egyidejűleg elérni. Ez a mutató gyors lehetőséget biztosít különböző hitelesítőrendszerek megbízhatóságának összehasonlítására.

Azokban az esetekben, amikor EER pontosan nem meghatározható, de a különbség FAR és FRR között elég kicsi, az EER egy elfogadható közelítése az Average Error Rate (AER). AER értékét, adott adatkészlet és határérték szerint számított, FAR és FRR átlagát véve számíthatjuk ki. A munkám során elért eredmények jellemzésére én is az átlagos hibaarányt használom.

2.5 Adatbázisok

Számos publikus online aláíradatbázis létezik, például az Online aláíráshitelesítés jelentősége című fejezetben említett nemzetközi versenyek hozadékaként, melyek használata elősegíti az irodalom eredményeinek összehasonlíthatóságát. További előnyt jelenthet ezeknek az alkalmazása, mivel európai és keleti országokban felvett aláírások egyaránt rendelkezésre állnak. Előfordulhat olyan, hogy egy rendszer egészen más pontossággal működik a különböző eredetű aláírásokon, és míg európai aláírásokra egészen nagy bizonyossággal hoz helyes eredményeket, addig a jelentősen eltérő keleti aláírásokon tesztelve már messze nem arat akkora sikert. [4] [5]

Az eredmények értékeléséhez fontos szempont, hogy az adatbázis felépítését illetően is rendelkezünk információval. Az adatbázisok leírásában ezért pontosan

meghatározzák, hogy milyen eszközt használtak az aláírások rögzítésére, profi (skilled) hamisítványokat vagy más aláírók eredeti aláírásait (random) használták-e. Továbbá, hogy hány aláíró vett részt és hozzájuk hány eredeti és hamis aláírást rendeltek, valamint rendszerint azt is, hogy ennek kivitelezését hány alkalommal végezték és milyen időközzel. Ezen felül, hogy pontosan milyen aláírástulajdonságokkal (X, Y, nyomás stb.) írhatjuk le az így kapott aláírásokat.

Az 1. táblázat a mérések során használt hat adatbázis felépítését mutatja be röviden.

Név	Aláírók száma	Eredeti aláírások száma		Hamis aláírások száma		Több információ
		Aláírónként	Össz.	Aláírónként	Össz.	
SVC2004	40	20	800	20	800	[2]
MCYT-100	100	20	2 000	25	2 500	[32]
Dutch	64	–	1 626	–	730	[4]
German	30	–	450	–	300	[6]
Chinese	20	–	475	–	891	[4]
Japanese	11	42	462	36	396	[5]

1. táblázat: Mérések során használt adatbázisok

3 Korábbi kutatási tevékenységem

Már régebb óta foglalkozom online aláíráshitelesítéssel, a Villamosmérnöki és Informatikai Kar 2018. évi TDK konferenciájára „On-line aláírás-hitelesítési algoritmusok vizsgálata” [33] címmel adtam be a kutatásomat összefoglaló dolgozatot. Jelen dolgozatom épít a korábbi kutatásomra, így a tavalyi dolgozatomra is bizonyos pontokon. Ebben a fejezetben a 2018. évi dolgozatom tartalmát szeretném röviden ismertetni.

3.1 Osztályozók típusai

Két merőben eltérő megközelítést alkalmaztam a hitelesítési döntés meghozataláért felelős osztályozók megvalósítása során. A jelen dolgozatban bemutatott mérések során is lényegében változatlanul ezt a két megközelítést alkalmaztam. Mindkettőben közös vonás, hogy természetesen használ eredeti referencia-aláírásokat, valamint, hogy két fázisra bontható a működésük.

Az első úgynevezett tanulási fázis célja meghatározni azt a határértéket, ami a későbbi döntés egyik alapját képezi. A második vagyis tesztelési fázisban az osztályozók olyan eredeti vagy hamis aláírást kapnak, aminek eredetéről nincs információjuk. Ez az ismeretlen eredetű aláírás összehasonlításra kerül a rendelkezésre álló referencia-aláírásokkal, majd az összehasonlítás eredményeként meghatározott távolságérték és a tanulási fázisban definiált határérték alapján meghozza a döntését a rendszer. Valójában azonban az ismeretlen eredetűnek tekintett aláírásokról is meg van az az információ, hogy eredeti-e, így az osztályozó döntésének helyességét azonnal ellenőrizni is lehet. A referenciákhoz történő hasonlítás során az egyes referenciákhoz viszonyított távolságok átlaga adja a végső távolságértéket.

Klasszikus esetben az osztályozók egyetlen bináris kimenetet szoktak eredményül adni. Amennyiben az ismeretlen aláírás referenciáktól mért távolsága nem nagyobb a határértéknél, akkor eredetinek, míg ellenkező esetben hamisnak tekinti az osztályozó. Az általam használt osztályozók ezzel szemben a határérték és a meghatározott távolság különbségével arányos, 0 és 1 közötti tartományba eső értéket képesek meghatározni, ami a döntés biztosságát hivatott jelezni. Amennyiben a határérték és a referenciáktól vett távolság éppen egyenlő, abban az esetben az osztályozó 0,5-öt ad eredményül. Ebben a

megközelítésben az osztályozó 0,5-nél kisebb értékű kimenete jelenti azt a döntést, hogy az aláírás hamis, az annál nagyobb pedig, hogy az aláírás eredeti.

Továbbá mindkét osztályozótípus aláírófüggő határértéket használ, ami azt jelenti, hogy a határérték meghatározása aláírónként történik. Ennek következtében az egyes aláírók esetén lehetőség van eltérő határértékek megválasztására is. Másik lehetőség az aláírófüggetlen határérték alkalmazása lenne, ami az egész adatbázisra vonatkozóan egy globális érték meghatározását jelenti. Az irodalomból egyértelműen kiderül, hogy az aláírófüggő megközelítés alkalmazása sokkal hatékonyabb [7], [18]. Ennek megfelelően munkám során én is minden esetben ezt használtam.

3.1.1 Minimális AER elérését célzó osztályozó

Ennek a megközelítésnek a célja, hogy meghatározza azt a minimális átlagos hibaarányt, amely adott konfiguráció mellett elérhető. A konkrét megvalósításban ez azt jelenti, hogy a tanulási fázis során a határérték úgy kerül meghatározásra, hogy a rendelkezésre álló tesztaláírásokon a lehető legjobb eredményt adja. Ebből következik, hogy a tesztaláírások már a tanulási fázisban ismertek és felhasználásra kerülnek a határérték megállapítása során. Emiatt ez a megközelítés gyakorlati megoldásokban nem használható, hiszen az ismeretlen eredetű aláírásokról valóban nem tudjuk az eredetüket, valamint tipikusan a tanulási fázis során még nem is állnak rendelkezésre.

Fontos megemlíteni azonban, hogy a gyakorlati alkalmazhatóság hiányában is nagy jelentőséggel bír ez az osztályozó is. Kutatási szempontból nagy segítséget nyújt különböző algoritmusok összehasonlításában, melyek egymás konkurensei lehetnek. Ugyanazt a tesztaláírást használva az elérhető elméleti minimum segítségével jól összehasonlíthatók lesznek a különböző módszerek, és ez alapján különböző megfigyeléseket tehetünk, melyek a későbbiekben hatékony, a gyakorlatban is alkalmazható rendszerek tervezését segíthetik. Továbbá az így elért eredmények jó viszonyítási alapot nyújtanak a valós körülmények közt is alkalmazható osztályozók által elért eredmények értékeléséhez.

Jelen dolgozatban egy ehhez nagyon hasonló működésű osztályozót alkalmaztam annyi különbséggel, hogy a tanulási fázisban nem a minimális AER elérése volt a cél, hanem az Equal Error Rate (EER) meghatározása.

3.1.2 Automatikus osztályozó

A jelen munkámban is alkalmazott automatikus osztályozó működéséhez a korábbiakkal ellentétben már semmilyen előismeretre nincsen szükség a tesztalíráásokat illetően. Ez a megközelítés a tanulási fázisban kizárólag az eredeti referencia-alíráásokra hagyatkozik. Ennek köszönhetően ez az osztályozó már valós alkalmazásokban is használható.

A tanulási fázisban az egyes referencia-alíráások a tőlük különböző összes másikkal összehasonlításra kerülnek, végül a korábbiaknak megfelelően az alírást az így mért távolságok átlagával jellemeztem. Az általam definiált automatikus osztályozó megalkotása során törekedtem az egyszerűsége, így a határérték meghatározása a mért átlagtávolságok átlaga és szórása alapján történik. Ez egyben azt is jelenti, hogy a módszer további finomításával az eredmények valószínűleg tovább javíthatók.

A második fázisban az eredeti és hamis tesztalíráások segítségével kerül ellenőrzésre, hogy a referenciák átlagtávolsága alapján a fentiek szerint meghatározott határérték mennyire eredményesen használható.

3.2 Alíráások összehasonlítására alkalmazott algoritmus

Ebben a fejezetben az osztályozók által az alíráások összehasonlítására, a távolságértékek meghatározására használt, a jelen kutatásom szempontjából releváns algoritmust szeretném bemutatni. Eredetileg két algoritmust implementáltam és vizsgáltam meg, melyek közül egyik maga a korábbiakban bemutatott DTW algoritmus volt. Az idei dolgozatomban bemutatott munka során kizárólag a DTW algoritmust alkalmaztam, mivel a céloom az előfeldolgozás hitelesítési pontosságra gyakorolt hatása volt, nem pedig a különböző algoritmusoké.

3.2.1 DTW algoritmus megvalósítása

A DTW algoritmus megvalósítása tulajdonképpen nem más mint, egy dinamikus programozási feladat, ahol a két idősor összehasonlítására egy mátrixot kell kitöltenünk. A mátrix sorai és oszlopai a két összehasonlítandó idősor, esetünkben alíráás közül az egyik, illetve a másik pontjait jelölik. A mátrix egyes celláinak értéke azt a minimális össztávolságot határozza meg, mely addig a pontig a két vektor között fennáll. Az első lépés a mátrix inicializálása, vagyis annak első sorának és oszlopának a kitöltése. A táblázat első (1,1) cellája a két idősor első pontjai közt mért távolságot tartalmazza. Az

első sor további mezőiben a cella által meghatározott két pont közti távolság és a mező bal szomszédja által tartalmazott érték összege áll. Hasonló módon kerül kitöltésre az első oszlop is, azzal a különbséggel, hogy a felső szomszéd értékét használjuk fel.

A mátrix további pontjait is szintén egy-egy kéttagú összeggel tudjuk meghatározni, mely kiszámításához szintén felhasználjuk a kitöltendő mező által meghatározott pontok közti távolságot. Az összeg másik tagját három, korábban kiszámolt, a kitöltendő mezővel szomszédos mezők ismeretében tudjuk meghatározni. Ennek a három mezőnek az értékei közül a legkisebbet választjuk az összeg második tagjának. Mindez formálisan a 2. ábra által használt elnevezések szerint $D = d(r_4, s_5) + \min \{A, B, C\}$.

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

2. ábra: DTW költségmátrixa

Az algoritmus során kitöltött költségmátrix a 2. ábra szerint N betűvel jelölt mezője tartalmazza a DTW algoritmus eredményét. Ennek értéke azt határozza meg, hogy mi az a minimális össztávolság, ami a két idősor pontjai között fennáll. Esetünkben ezt az értéket használjuk két aláírás távolságának meghatározására.

A jelen dolgozatban bemutatott mérések során alkalmazott osztályozók minden esetben DTW algoritmust alkalmaznak az aláírások összehasonlítására, mivel a munkám ez esetben nem a különböző összehasonlítási megoldásokra fókuszál.

3.3 Alírástulajdonságok megválasztása

Korábbi dolgozatomban azt is vizsgáltam, hogy a hitelesítés során használt alírástulajdonságok megválasztása milyen hatással van az eredményekre. A méréseket kizárólag az SVC2004 Task2 adatbázison végeztem. A tulajdonságokat két nagy csoportra bontottam, így alaptulajdonságokat és származtatott tulajdonságokat különböztettem meg. Az alaptulajdonságok halmazát az általam használt adatbázisban rögzített tulajdonságok adták. A származtatott tulajdonságok ezzel szemben nem állnak közvetlenül rendelkezésre az adatbázisban, azonban az alaptulajdonságokból levezethetők valamilyen módon.

3.3.1 Tulajdonságok kombinációi

A DTW algoritmus képes tulajdonságok egy halmazát is kezelni az osztályozás során. Ilyenkor az aláírás egy pontját nem egy-egy a választott tulajdonsághoz tartozó értékkel jellemezzük, hanem mindegyik ponthoz egy-egy vektort rendelünk a használt halmaznak megfelelő értékekkel. A vektorok meghatározott sorrendben tartalmazzák a vizsgált tulajdonság-halmaz egyes elemeihez rögzített értékeit az aláírásnak. Ebben az esetben a DTW algoritmus a pontokhoz rendelt vektorok egymáshoz viszonyított távolságát használja fel az aláírások közti távolságok meghatározása során.

A munkám során nem vizsgáltam meg az összes előállítható kombinációját a tulajdonságoknak, azonban számos lehetőséget megvizsgáltam. A dolgozatban két-, három- és négyelemű halmazokkal is foglalkoztam. Jelen dolgozat számos mérési konfigurációja is tulajdonsághalmazokat alkalmaz.

3.4 Előfeldolgozás

A korábbi dolgozatomban az előfeldolgozás egy nagyon kis szegmense jelent meg, mindössze két különböző előfeldolgozás eredményekre gyakorolt hatását hasonlítottam össze egymással és az előfeldolgozás hiányával. A két megoldást a dolgozatban normalizálás és súlypontba igazítás néven mutattam be.

A normalizálást a különböző tulajdonságok értéktartományának $[0,1]$ tartományba való leképezéseként definiáltam. Ez tulajdonképpen egy skálázást jelentett, aminek célja a különböző tulajdonságok értékeinek egymással összemérhetővé tétele volt. A súlypontba igazítás az aláírások egyfajta eltolásaként definiáltam, aminek első

lépése az aláírás súlypontjának meghatározása volt. Ezután került sor az aláírások igazítására oly módon, hogy a súlypontok egymás fölé, egy tengely mentére kerüljenek.

Dolgozatomban a célom nem az előfeldolgozás lehetőségeinek részletes vizsgálata és összehasonlítása volt, sokkal inkább a származtatott tulajdonságok használatának és az előfeldolgozó lépések viszonyának feltérképezése. Ebből kifolyólag megvizsgáltam a kettő sorrendjének szerepét is.

4 Előfeldolgozási algoritmusok

Ebben a fejezetben az általam vizsgált megközelítésekhez tartozó általam implementált algoritmusok kerülnek bemutatásra, melyeken a mérések elvégzése is történt. Az algoritmusok két nagy csoportba sorolhatók.

Az első csoportot az újramintavételezési algoritmusok alkotják, melyek az aláírások begyűjtése során rögzített pontok számosságára is hatással vannak. Ezeknek a lényege a rögzítéskor használt mintavételezési beállítások módosítása, mely mintavételi pontok törlésével és új pontok felvételével egyaránt járhat.

A másik csoportba tartozó algoritmusok ezzel szemben nincsenek hatással az aláírásokat leíró pontok számára, csupán az egyes pontokhoz rendelt különböző tulajdonsáértékekre. Ide sorolandók az elforgatás, eltolás és skálázás, melyek mind az aláírások lokációjából fakadó problémákat hivatottak orvosolni.

Az algoritmusok bemutatása előtt szükség van néhány jelölés definiálására. Egy S online aláírást a hozzá tartozó tulajdonságok segítségével a következő módon tudjuk leírni: $S = \{F_1, F_2, F_3, \dots, F_m\}$, ahol $F_i = \{v_1, v_2, v_3, \dots, v_n\}$ az F_i tulajdonsághoz rögzített v_j értékeket tartalmazza és a rögzített pontok száma n .

4.1 Újramintavételezési algoritmusok

Az újramintavételezési algoritmusok tehát rögzített pontok törlését és/vagy új pontok meghatározását eredményezik. Az új pontok meghatározása a vizsgált aláírástulajdonságok értékeinek a rögzített pontok alapján interpolációval történő kiszámítását jelenti. Ennek megfelelően elsőként az általam alkalmazott két interpolációs megoldást fogom bemutatni, melyeket az újramintavételezési algoritmusokban egységesen alkalmaztam új pontok meghatározása esetén.

4.1.1 Interpoláció

Az interpoláció megvalósítására lineáris (linear) és harmadfokú (cubic) interpolációt alkalmaztam.

4.1.1.1 Lineáris interpoláció

Az interpolációs algoritmus bemenetként megkapja azt a t_g időbélyeget, amihez az új generált pontot rögzíteni szeretnénk. Az első lépés lineáris interpoláció esetén, hogy megtaláljuk a két pontot meghatározó időbélyegeket, melyek közé az új pontot szeretnénk beszúrni. Amennyiben a beszúrandó időpillanathoz már tartozik rögzített pont, vagyis $t_g \in T = \{t_1, t_2, t_3, \dots, t_n\}$, akkor nincs szükség interpolációra.

Ellenkező esetben, ha $t_g \notin T = \{t_1, t_2, t_3, \dots, t_n\}$, az azt jelenti, hogy létezik az időbélyegek között két olyan érték, $t_i, t_{i+1} \in T = \{t_1, t_2, t_3, \dots, t_n\}$, melyekre igaz, hogy $t_i < t_g < t_{i+1}$. Jelölje ekkor v_i és v_{i+1} a vizsgált tulajdonság t_i és t_{i+1} időpillanatokhoz rögzített értékét.

Ez alapján a vizsgált tulajdonság új ponthoz tartozó v_g értékének kiszámítása a következőképpen történik:

$$v_g = v_i + \frac{(t_g - t_i) \times (v_{i+1} - v_i)}{t_{i+1} - t_i} \quad (1)$$

4.1.1.2 Harmadfokú interpoláció

Az első lépés ebben az esetben is a beszúrandó t_g időbélyeg helyének megtalálása. Amennyiben a beszúrandó időpillanathoz már tartozik rögzített pont, vagyis $t_g \in T = \{t_1, t_2, t_3, \dots, t_n\}$, akkor továbbra sincs szükség interpolációra.

Abban az esetben viszont, ha $t_g \notin T = \{t_1, t_2, t_3, \dots, t_n\}$, akkor meg kell találni azokat a $t_i, t_{i+1} \in T = \{t_1, t_2, t_3, \dots, t_n\}$ időbélyegeket, melyek közé az új pontot be kell szúrni, vagyis $t_i < t_g < t_{i+1}$. Harmadfokú interpoláció esetén azonban az új tulajdonságérték kiszámításához nem elég csak a két szomszédos pont, szükség van a kettővel előbbi és kettővel utána következő pontokra is. Mindez formálisan azt jelenti, hogy $t_{i-1}, t_i, t_{i+1}, t_{i+2} \in T = \{t_1, t_2, t_3, \dots, t_n\}$ időbélyegekre és a hozzájuk tartozó $v_{i-1}, v_i, v_{i+1}, v_{i+2}$ értékekre van szükség.

Amennyiben a generálandó pont az első vagy utolsó két pont között fekszik, akkor t_{i-1} és t_{i+2} időbélyegek nem értelmezhetők. Ebben a kivételes esetben az algoritmus a korábban bemutatott lineáris interpolációt alkalmazza. Normál esetben a generált ponthoz tartozó v_g tulajdonságérték a következő képletek segítségével számítandó.

$$t = \frac{t_g - t_i}{t_{i+1} - t_i} \quad (2)$$

$$h_{00} = 2 \times t^3 - 3 \times t^2 + 1 \quad (3)$$

$$h_{10} = t^3 - 2 \times t^2 + t \quad (4)$$

$$h_{01} = -2 \times t^3 + 3 \times t^2 \quad (5)$$

$$h_{11} = t^3 - t^2 \quad (6)$$

$$m_0 = \frac{1}{2} \left(\frac{v_{i+1} - v_i}{t_{i+1} - t_i} + \frac{v_i - v_{i-1}}{t_i - t_{i-1}} \right) \quad (7)$$

$$m_1 = \frac{1}{2} \left(\frac{v_{i+2} - v_{i+1}}{t_{i+2} - t_{i+1}} + \frac{v_{i+1} - v_i}{t_{i+1} - t_i} \right) \quad (8)$$

$$v_g = h_{00} \times v_i + h_{10} \times (t_{i+1} - t_i) \times m_0 + h_{01} \times v_{i+1} + h_{11} \times (t_{i+1} - t_i) \times m_1 \quad (9)$$

4.1.2 Újramintavételezés a pontok fix száma alapján

Ez a megközelítés a mintavételi gyakoriság egységesítésére szolgál. Az a kiinduló gondolat, hogy létezhetnek olyan eszközök, melyek a rögzítés folyamán nem képesek tartani az egyenlő időközöket két pont között, így azt eredményezik, hogy a két pont rögzítése közt eltelt idő egy bizonyos tartományban mozogva változó értékeket vesz fel. Másik gyakori jelenség, hogy az aláírások nem ugyanannyi pontból állnak. Utóbbit egyébként a DTW algoritmus gond nélkül képes kezelni, hiszen működésének nem feltétele, hogy az összehasonlítandó idősorok egyenlő hosszúak legyenek. Ez az újramintavételezési megoldás mindkét említett jelenségre egyszerre nyújt megoldást.

Az algoritmus számára meg kell adni, hogy hány pontot tartalmazzon az aláírás az újramintavételezés után. Jelölje ezt a számot n^* , míg az aláírás pontjainak eredeti számát jelölje n . Első lépésben az új egységes t időköz kerül meghatározásra, mely minden két pont között fenn fog állni. Ennek kiszámítása az aláírás eredeti hossza, vagyis a legnagyobb és legkisebb időbélyegek különbsége, valamint a meghatározott n^* alapján történik. Legyenek az eredeti időbélyegek, ekkor t a következőképp definiálható:

$$t = \frac{t_n - t_1}{n^* - 1} \quad (10)$$

Ennek ismeretében az újramintavételezés után használt $T' = \{t'_1, t'_2, t'_3, \dots, t'_{n^*}\}$ időbélyegek meghatározása következik az első pont $t'_1 = t_1$ értékadásától kezdődően

$$t'_i = t'_{i-1} + t \quad (11)$$

szerint. Az új időbélyegek között lehetnek olyanok is, amelyek az eredeti halmazban is megtalálhatók voltak, de jellemzően újak keletkeznek.

Végső soron a konfigurációnak megfelelő interpoláció felhasználásával kerül sor az új időbélyegekhez tartozó tulajdonságértékek kiszámítására.

4.1.3 Nulla nyomású pontok eltávolítása

A cél egységesen megszüntetni az aláírások tollfelemelések alatt rögzített pontjainak figyelembevételét. A nulla nyomású pontok eltávolítása szemléletesen az aláírások összehúzását jelenti a hézagoknál, ahol nem ért papírt a toll. Ez az algoritmus azonban nem csak az aláírás közbenső pontjain alkalmazható, hanem az aláírás elején vagy végén rögzített zaj eltávolítására is alkalmas. Utóbbinak akkor lehet nagy jelentősége, ha a rögzítés az aláírás megkezdése előtt és annak befejezése után is folyamatban van, így a toll kezdőpozícióba történő behúzását, majd végpontból történő kivezetését is az aláírás részeként értelmezi.

Az algoritmus működése egészen egyszerű. Jellemezzon egy S aláírást a nyomást is tartalmazó $S = \{P, F_1, F_2, F_3, \dots, F_m\}$ tulajdonsághalmaz. Az aláírás j . pontjának eltávolítására akkor kerül sor, ha a $P = \{p_1, p_2, p_3, \dots, p_j, \dots, p_n\}$ nyomásértékek j . aláírásponthoz tartozó p_j eleme nulla. Ebben az esetben minden F_i tulajdonsághoz rögzített értékhalmból törlésre kerül v_j értéke.

4.1.4 Tollfelemelések kitöltése

A tollfelemelésének kezelésére egy másik lehetőség, hogy az alattuk rögzített pontokat is figyelembe vesszük az aláírások közti távolság meghatározásakor. Előfordulhat azonban, hogy bár emellett a megközelítés mellett döntünk, a rendelkezésre álló aláírások nem tartalmaznak információt a tollfelemelések idejéről. Ebben az esetben nyújthat megoldást a tollfelemelések kitöltése.

Az alaptulajdonságok között idő tekintetében kizárólag az egyes pontokhoz tartozó időbélyegek szerepelnek, melyek csak az adott pillanat jellemzésére szolgálnak, nem pedig az idő múlására. A szomszédos pontok közt eltelt idő a pontokhoz tartozó időbélyegek alapján könnyen kiszámítható. Az egyértelműség érdekében az eltelt idők hossza mellett tároljuk a szomszédos pontokat meghatározó időbélyegeket is, amik alapján a számítást végeztük. Az így kialakuló hármas felhasználásával egyértelműen hivatkozhatunk bármely két szomszédos pont mintavételezése közt eltelt időre.

Az idő múlásának kezelésére vezessünk be egy új tulajdonságot időköz néven, melyet az említett hármas segítségével lehet jellemezni. Jelölése legyen $ts_i = \{t_0, t_1, l\}$,

ahol t_0 és t_1 a két szomszédos ponthoz tartozó időbélyegek, amik a köztük lévő időintervallum első és utolsó időbélyegeit adják, l pedig jelölje ezek különbségét, vagyis az időköz hosszát.

Az algoritmus első lépéseként kiszámítja az újonnan bevezetett időköz tulajdonság $TS = \{ts_1, ts_2, ts_3, \dots, ts_{n-1}\}$ értékeit, majd ezzel a plusz információval kiegészíti az $S = \{T, F_1, F_2, F_3, \dots, F_m\}$ aláírás leírását. A ts_i értékek kiszámítása az eredeti $T = \{t_1, t_2, t_3, \dots, t_n\}$ időbélyegek alapján a következőképp történik: $ts_i = \{t_0, t_1, l\}$, ahol $t_0 = t_i$, $t_1 = t_{i+1}$ és $l = t_1 - t_0 = t_{i+1} - t_i$, továbbá $t_i, t_{i+1} \in T = \{t_1, t_2, t_3, \dots, t_n\}$.

Az időközök ismeretében az algoritmus következő lépése a tollfelemelések meghatározására szolgál. Egy időközt akkor nevezünk tollfelemelésnek, ha a hossza nagyobb, mint az időközök hosszának mediánja. Ennek megfelelően a tollfelemelések formális jelölése legyen $pd = \{t_0, t_1, l\}$, ahol $l > \tilde{l}$.

Az algoritmus utolsó lépését a megtalált tollfelemelések új pontokkal történő feltöltése jelenti. Ennek során $k = \lfloor l/\tilde{l} \rfloor$ generált pont kerül beszúrásra az aláírás $pd = \{t_0, t_1, l\}$ tollfelemelésének helyére. Az új pontok időbélyegét a következő egyenlet szerint határozom meg:

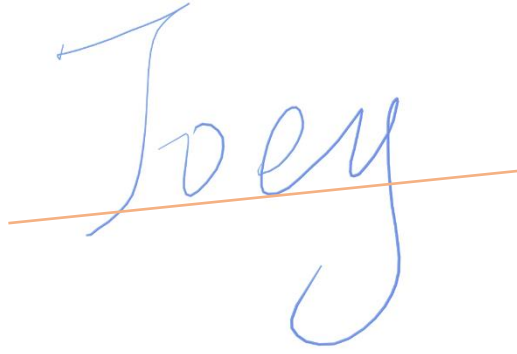
$$t_i = \begin{cases} t_0 + \tilde{l}, & i = 0 \\ t_{i-1} + \tilde{l}, & 0 < i < k \end{cases} \quad (12)$$

A többi tulajdonság új időbélyegekhez tartozó értékeit az algoritmus interpoláció segítségével határozza meg.

4.2 Elforgatás

Az elforgatás célja az aláírások egymáshoz viszonyított elfordulásának egységes eliminálása. Az aláírás elfordulása ugyan egy könnyen elképzelhető, egyszerűen szemléltethető fogalom, ám formális definíciót adni valamivel nehezebb. Ez mégis elkerülhetetlen az elforgatási algoritmus megvalósításának bemutatása.

Nevezzük elfordulásnak az aláírás alapvonala és a vízszintes által bezárt szöget. Az aláírás alapvonalát a 3. ábra illusztrálja. Az elforgatási algoritmus megvalósítása során az $S = \{X, Y, T, F_1, \dots, F_m\}$ aláírás X , Y és T tulajdonságait, vagyis a koordinátainformációkat és az időbélyegeket használtam fel.



3. ábra: Az aláírás alapvonala

Az alapvető elvárás az, hogy amennyiben az X- és Y-koordinátákat időfüggvényként kezeljük, akkor az X-értékek az idővel egyenletesen nőnek, míg az Y-értékek egy az átlagérték körüli fix intervallumon belül helyezkednek el. Ennek megfelelően az Y-értékek növekedése az aláírás elfordulására utalhat.

Az algoritmus első lépéseként az Y-értékekre, mint időfüggvényre közelítő egyenest illesztettem, és amennyiben ez nem vízszintes, akkor az aláírást elfordultnak tekintettem. Az illesztett egyenes ugyan nem egyezik meg az aláírás alapvonalával, azonban párhuzamos vele. Ebből kifolyólag a vízszintessel bezárt szögük egyenlő, így egymás helyettesítéseként lehet őket használni. A közelítő egyenest az $Y = \{y_1, y_2, y_3, \dots, y_n\}$ és $T = \{t_1, t_2, t_3, \dots, t_n\}$ tulajdonságok alapján a következő egyenletek szerint határoztam meg:

$$a = \frac{\frac{\sum_i t_i \times y_i}{n} - \bar{t} \times \bar{y}}{\frac{\sum_i t_i \times t_i}{n} - \bar{t} \times \bar{t}} \quad (13)$$

$$b = a \times \bar{t} - \bar{y} \quad (14)$$

$$y_{fitting_i} = a \times t_i - b \quad (15)$$

A vízszintes egyenes meghatározására $\min_i t_i$; $\max_i t_i$; $\min_i y_{fitting_i}$ értékeket használtam. Az elfordulás α szögét az illesztett egyenes és a vízszintes segítségével az alábbi egyenletnek megfelelően számoltam ki:

$$\alpha = \left| \tan^{-1} \left(\frac{\min_i y_{fitting_i} - \max_i y_{fitting_i}}{\min_i t_i - \max_i t_i} \right) - \tan^{-1} \left(\frac{\min_i y_{fitting_i} - \min_i y_{fitting_i}}{\min_i t_i - \max_i t_i} \right) \right| \quad (16)$$

Az algoritmus megvalósításának utolsó lépése maga az elforgatás, amely az eredeti koordinátainformációk módosítását jelenti. Az elforgatás során az x_i és y_i értékeket a (17) és (18) egyenlet szerint kiszámított x'_i és y'_i értékekre cseréltem.

$$x'_i = (x \times \cos \alpha) - (y \times \sin \alpha) \quad (17)$$

$$y'_i = (x \times \sin \alpha) + (y \times \cos \alpha) \quad (18)$$

4.3 Eltolás

Az aláírások különböző elhelyezkedése az elforduláshoz hasonlóan egy olyan probléma, ami kizárólag a koordinátainformációkat érinti. Az eltolással az aláírásokat valamely pontjuk alapján szeretnénk egymáshoz illeszteni. Ennek megvalósítása két lépcsőben történik, először meghatározom azt a pontot, ami alapján illeszteni szeretnénk, majd ezután kerül sor az aláírás megfelelő pozícióba való eltolására.

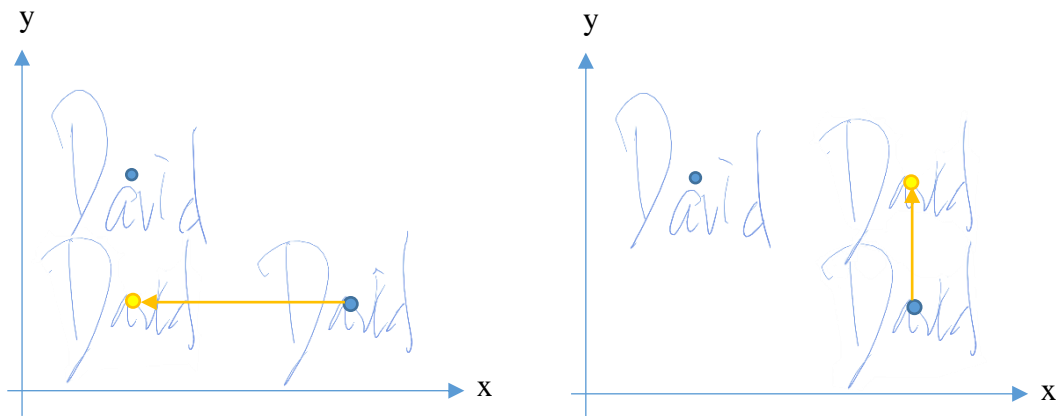
Az eltolás az X- és Y-értékek mentén egyaránt történhet, a kettő egymástól teljesen függetlenül kivitelezhető, így lehetőség van csak az egyik, csak a másik, vagy mindkettő szerint is elmozdítani az aláírást. Az eltolás előzőekben leírt iránya határozza meg az eltolás dimenzióját, vagyis, hogy az $X = \{x_1, x_2, x_3, \dots, x_n\}$, $Y = \{y_1, y_2, y_3, \dots, y_n\}$ tulajdonság vagy mindkettő használatát. A viszonyítási pont, vagyis az új origó meghatározása után, maga az eltolás egy egyszerű képlet segítségével megvalósítható. Az aláírás elmozdítása a gyakorlatban az eltolás dimenzióját meghatározó tulajdonság(ok) eredeti értékeinek módosítását jelenti. Legyen az eltolás formális definíciója

$$v'_i = v_i - o_{new}, \quad (19)$$

ahol v'_i az aláírás i . pontjának eltolt értéke, v_i ugyanennek a pontnak az eredeti értéke, o_{new} pedig az új origó.

Az új origó meghatározására számos különböző lehetőség van, én két fő megközelítést alkalmaztam. Az első esetben az eltolás dimenziójának megfelelő tulajdonság(ok) súlypontját választottam origónak. Ezt a megközelítést a lehetséges mindhárom módon megvalósítottam, ennek köszönhetően az előfeldolgozási lépések konfiguráció során ki lehet választani, hogy X, Y vagy mindkettő értékei szerint kerüljön kiszámításra az aláírás súlypontja. Az X és Y külön-külön való használata egy tengely menti eltolást eredményez, ahogyan azt a 4. ábra is szemlélteti. Az aláírás valódi súlypontjának használatát X és Y együttes használata jelenti. Ebben az esetben a két

egymástól független, egy tengely menti eltolás valósul meg egymás után, melyek sorrendje mindegy, függetlenségük miatt az eredményen nem változta.



4. ábra: X- és Y-értékek szerinti súlypontba való eltolás

Az aláírás súlypontját, mely egyben az új origót is jelenti, az $X = \{x_1, x_2, x_3, \dots, x_n\}$ tulajdonság alapján a következő képlettel határoztam meg:

$$o_{new} = \frac{\sum_i x_i}{n} \quad (20)$$

Hasonló módon, az aláírás Y-tengely menti súlypontjába igazításhoz az origót az alábbi módon számítottam ki:

$$o_{new} = \frac{\sum_i y_i}{n} \quad (21)$$

A másik fő megközelítés az aláírás szélsőértékeit használja viszonyítási pontként. Ez az előzőekhez hasonlóan szintén X, Y vagy mindkettő értékeinek alapján történhet, és az együttes alkalmazás esetén két egymástól független, egy tengely menti eltolás két lépcsőben történő elvégzésére kerül sor. A szélső értékek segítségével minden aláíráshoz meg lehet határozni egy befoglaló téglalapot, melynek minden oldala legalább egy pontban érinti az aláírást. Az érintési pontokban az $x = \min_i x_i$, $x = \max_i x_i$, $y = \min_i y_i$ és $y = \max_i y_i$ feltételek legalább egyike teljesül. A befoglaló téglalap segítségével könnyen szemléltethető a szélsőértékek szerinti eltolás, miszerint lényegében a befoglaló téglalap oldalai alapján történik az illesztés.

Az eltolás dimenziójának megválasztásának három különböző lehetősége összesen nyolc különböző konkrét megvalósítási megoldást eredményez. A befoglaló téglalap oldalait felhasználva négy olyan eset van, amikor az eltolás csak egy tulajdonság

mentén történik, és további négy olyan eset, amikor mindkét koordinátatulajdonság értékei módosulnak. A nyolc lehetőség közül én egyet választottam és csak azt valósítottam meg.



5. ábra: Alíírás befoglaló téglalapja

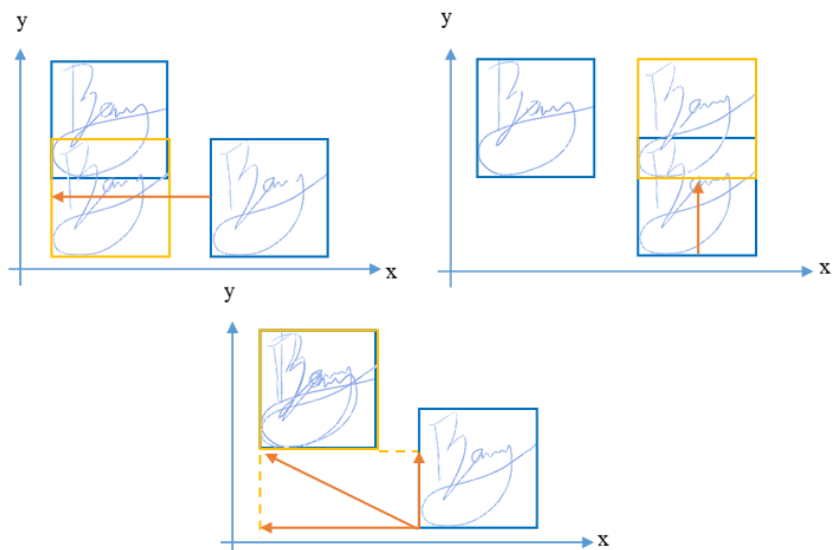
Az általam választott megoldás az alíírásokat a befoglaló téglalap bal alsó sarka szerint illeszti egymáshoz. Ezt úgy implementáltam, hogy az új origónak mindkét tulajdonság esetében azok minimumértékét. Az X-értékek esetén használt formula:

$$o_{new} = \min_i x_i \quad (22)$$

Hasonlóképp az Y tulajdonság új origóját az alábbi képlettel határoztam meg:

$$o_{new} = \min_i y_i \quad (22)$$

A bal alsó sarokba való illesztés során az X és Y tulajdonságokat egymástól teljesen függetlenül módosítottam, a téglalap bal és alsó oldala mentén végzett igazítások sorrendje teljesen mindegy az eredmény szempontjából. Az eltolási algoritmus befoglaló téglalap szerinti működését a 6. ábra illusztrálja.



6. ábra: Az alíírás befoglaló téglalapjának bal alsó sarka szerinti eltolás

Az eltolási algoritmus a mérések során alkalmazott megvalósításait és azok főbb jellemzőit a 2. táblázat foglalja össze.

Név	Tulajdonság 1	Tulajdonság 2	Origó meghatározásának alapja
BottomLeft	X	Y	szélsőérték
CogX	X	–	súlypont
CogY	Y	–	súlypont
CogXY	X	Y	súlypont

2. táblázat: A mérések során használt eltolási megoldások

4.4 Skálázás

Az aláírások eltérő mérete szintén elég szembetűnő különbségeket eredményezhet, melyek a gyakorlatban az eredeti aláírások egymásnak megfelelő pontjainak koordinátaértékei közt fennálló nagy távolságokat jelentik. Annak érdekében, hogy ezeket az osztályozás során használt DTW algoritmust félrevezető különbségeket megszüntessük, célszerű az aláírásokat egységes méretűre skálázni.

Az általam használt skálázás definíció szerint az aláíráshoz tartozó rögzített értékek adott $[a, b]$ intervallumba való leképezését jelenti. Az előző két algoritmusnál látottakkal teljesen megegyezően, a skálázást is lehetőségünk van csak az X-, csak az Y-tengely vagy mindkettő mentén végrehajtani. Az általam megvalósított skálázó algoritmust csak az aláírások koordinátatulajdonságain alkalmaztam a mérések során, azonban minden lehetőség adott ezt más tulajdonságokra is kiterjeszteni. Éppen ezért az alábbiakban egy tetszőleges $F = \{v_1, v_2, v_3, \dots, v_n\}$ tulajdonságra fogom meghatározni az értékek $[a, b]$ intervallumba történő leképezését. A választott tulajdonság eredeti v_i értékét a következő egyenletek segítségével kiszámított v'_i értékre cseréltem a skálázás során.

$$v_{min} = \min_i v_i \quad (23)$$

$$v_{max} = \max_i v_i \quad (24)$$

$$v'_i = a + \frac{(v_i - v_{min}) \times (b - a)}{v_{max} - v_{min}} \quad (25)$$

Az ilyen módon történő skálázás egymástól függetlenül módosítja az X és Y tulajdonságok értékeit akkor is, ha egyszerre alkalmazzuk őket. Az összetett esetben is két teljesen különálló és független skálázás történik a két tengely mentén. A mérések során

a bemutatott skálázási algoritmust mindig fixen a [0,1] intervallumra alkalmaztam, ez azonban nem biztosítja minden esetben az eredeti arányok megtartását.

Ennek a problémának a kiküszöbölésére definiáltam a skálázás egy egységes, aránytartó változatát. Az aránytartó skálázás lényege, hogy az algoritmus egyszerre két tulajdonság értékeit is módosítja az egyszerű skálázási algoritmusommal szemben. Ennek megfelelően az aránytartó skálázási algoritmust úgy valósítottam meg, hogy működéséhez az intervallum mellett két tulajdonság megadására van szüksége.

Legyen a leképezés céltartományát képező intervallum továbbra is $[a, b]$. Az ebbe a tartományba leképezendő tulajdonságot nevezzük alaptulajdonságnak és jelölése legyen $BF = \{bf_1, bf_2, bf_3, \dots, bf_n\}$. A másik tulajdonság jelölése $PF = \{pf_1, pf_2, pf_3, \dots, pf_n\}$ legyen és tartozzon hozzá egy $[a', b']$ intervallum, amibe az leképezésre kerül. Az utóbbi intervallum kezdetét meghatározó a' előre definiálható, akár csak a korábban látott a és b értékek, de b' értékét az algoritmus fogja kiszámítani az alaptulajdonság skálázásával arányosan. Az ezekkel a paraméterekkel rendelkező aránytartó skálázási algoritmusom futása során az új értékeket a következő formulák segítségével számítom ki:

$$bf_{min} = \min_i bf_i ; pf_{min} = \min_i pf_i ; bf_{max} = \max_i bf_i ; pf_{max} = \max_i pf_i \quad (26)$$

$$b' = a' + \frac{b-a}{bf_{max}-bf_{min}} \times (pf_{max} - pf_{min}) \quad (27)$$

$$bf' = a + \frac{(bf_i - bf_{min}) \times (b-a)}{bf_{max}-bf_{min}} \quad (28)$$

$$pf' = a' + \frac{(pf_i - pf_{min}) \times (b'-a')}{pf_{max}-pf_{min}} \quad (29)$$

A mérések során öt különböző skálázási megoldást hasonlítottam össze, melyek minden esetben $a = 0, b = 1, a' = 0$ paraméterekkel dolgoztak. A módszerek nevét és a további lényeges konfigurációs paraméterekre vonatkozó információkat a 3. táblázat összegzi. Aránytartó skálázás esetén a táblázat „Tulajdonság 1” oszlopa jelöli az alaptulajdonságot

Név	Tulajdonság 1	Tulajdonság 2	Aránytartó-e
X01	X	–	–
Y01	Y	–	–
X01Y0prop	X	Y	igen
Y01X0prop	Y	X	igen
X01Y01	X	Y	nem

3. táblázat: A mérések során használt skálázási megoldások

5 Egyéb hitelesítési paraméterek

A futtatandó mérések konfigurációjában természetesen meg kell határozni, hogy az előző fejezetben bemutatott előfeldolgozó lépések közül melyeket szeretnénk alkalmazni. A használt eltolási és a skálázási megoldásoknak vannak olyan lehetséges kombinációi, melyek együttes használata nem értelmes, így ezek kizárására figyelmet kell fordítani.

Az előfeldolgozó lépések végrehajtása után kerülhet sor magára az osztályozásra, ennek megfelelően pedig a használandó osztályozó típusát is meg kell paraméterként adni, amit két lehetőség közül kell kiválasztani. Az egyik ilyen a DTW alapú automatikus osztályozó, a másik pedig a szintén DTW alapú EER elérést célzó, vagy másnéven optimális osztályozó. A típus megválasztásán túl, a DTW algoritmus által használt távolságfüggvény tekintetében is döntést kell hozni, mely során az Euklideszi- és Manhattan-távolságok közül lehet választani.

A hitelesítési szempont, vagyis az osztályozás során vizsgált tulajdonságokat is előre meg kell határozni. A jelen dolgozatomban bemutatott mérések során kizárólag az alaptulajdonságok kerültek felhasználásra, azok viszont önmagukban és többemű halmaz részeként egyaránt. A hitelesítési szempont mérések során vizsgált konfigurációit a 4. táblázat egyes sorai tartalmazzák, míg a rendelkezésre álló alaptulajdonságok halmaznak egészét az oszlopok határozzák meg. A táblázat egyes mezőiben „+” jelöli, ha az adott konfiguráció használja az oszlopnak megfelelő tulajdonságot, és hasonlóan „-” jelöli, amennyiben nem használja.

Név	X	Y	P	Altitude	Azimuth
Altitude	-	-	-	+	-
Azimuth	-	-	-	-	+
P	-	-	+	-	-
X	+	-	-	-	-
XY	+	+	-	-	-
XYP	+	+	+	-	-
XYPAzimuthAltitude	+	+	+	+	+
Y	-	+	-	-	-

4. táblázat: Vizsgált aláírástulajdonságok tekintetében alkalmazott konfigurációk

6 Mérési keretrendszer

A mérések a SigStat [34] aláírás-hitelesítést támogató keretrendszer segítségével történtek, melynek a fejlesztői közt vagyok én is. A mérések futtatására a Superman [35] környezetben került sor, mely a BME kutatási célokra alkalmazható szuperszámítógépe. A 30 számítógépségből álló rendszer a Műegyetem alapkutatásainak támogatásához nyújt infrastruktúrát.

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. Az eredményeket különböző, beszédesen elnevezett .xlsx kiterjesztésű táblázatok formájában kapja a felhasználó, melyek tartalmazzák az aláíronként elért átlagos FAR, FRR és AER értékek, valamint a teljes adatbázisokon 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.

Az összes vázolt különböző megközelítés és egymást ki nem záró kombinációinak megadására van mód. Meg lehet határozni a használandó előfeldolgozó algoritmusokat, az osztályozó típusát, a hitelesítési szempontot, a használandó távolságfüggvényt és a referenciák mintavételezési megközelítését is.

6.1 Elvégzett mérések

Az előfeldolgozó lépések hatásainak összehasonlítására, ezekről széles körben helytálló megfigyelések alátámasztása céljából a rendszer ugyanazon konfigurációit mind a hat korábban bemutatott adatbázison (SVC2004, MCYT-100, Dutch, German, Chinese, Japanese) futtattuk. A benchmark konfigurációjának így részét képezi a felhasznált adatbázis meghatározása is. Ez egy nagyon kiterjedt, több lépcsős mérésorozat kivitelezését jelentette, melyben a konzulensem és néhány hallgatókolléga is a segítségemre volt.

A mérésorozat összesen 217 088 különböző benchmark futtatását ölelte fel, ami 70 GB mérési adatot eredményezett. A mérések futtatása 5 napot vett igénybe a Superman

számára, majd további több mint 12 óra kellett az adatok letöltéséhez és összegzéséhez. Ezután tudtam az eredményeket tételesen megvizsgálni, és a következő fejezetben bemutatott megfigyeléseket megtenni.

7 Mérési eredmények

Ebben a fejezetben mérési eredményeket, valamint ezek alapján tett néhány megfigyelésemet szeretném bemutatni, majd röviden értékelni mindezt. Az eredményeket a megfigyelések erősrendjét követve fogom bemutatni.

7.1 Pontok fix száma alapján végzett újramintavételezés hatása

A pontok fix száma alapján történő újramintavételezési algoritmust négy különböző értékkel (50, 100, 500, 1000) is teszteltük. Ezen értékek arra törekedve kerültek meghatározásra, hogy a vizsgált adatbázisok által az aláírások leírására használt pontok átlagos számát figyelembe véve, minden aláírásra legyen olyan konfiguráció, ami növeli és ami csökkenti az eredeti pontok számát.

Az eredmények alapján egyértelmű, hogy nem éri meg alkalmazni ezt az újramintavételezési megoldást. Az alábbi táblázatok azt mutatják meg adatbázisonként, hogy az algoritmus adott paraméterrel történő alkalmazása az esetek hány százalékában rontott a hitelesítés pontosságán az algoritmus elhagyásával elért eredményekhez képest. A negatív hatás megállapítására minden esetben két benchmark konfigurációt vettem össze, melyek egyetlen paraméter tekintetében különböztek csak, ami nem más, mint a pontok fix száma alapján végzett újramintavételezés. Az újramintavételezés nélkül és az újramintavételezéssel elért átlagos hibarányokat összehasonlítva megállapítottam, hogy az algoritmus használata pozitív, negatív vagy semleges hatással bírt-e. A táblázat a negatív eseteknek arányát tartalmazza az összes pár számának fényében. Az 5. táblázat az EER elérését célzó, optimális osztályozó, míg a 6. táblázat az automatikus osztályozó által elért eredményeket tartalmazza.

Adatbázis	50	100	500	1000
SVC2004	98,44%	100,00%	100,00%	100,00%
MCYT-100	100,00%	99,88%	99,98%	99,95%
Dutch	100,00%	100,00%	100,00%	100,00%
German	99,96%	100,00%	100,00%	100,00%
Chinese	100,00%	100,00%	100,00%	100,00%
Japanese	84,02%	74,10%	97,11%	100,00%

5. táblázat: Fix pont alapú újramintavételezés optimális osztályozó pontosságára gyakorolt negatív hatásának arányai

Adatbázis	50	100	500	1000
SVC2004	99,98%	99,95%	99,95%	99,95%
MCYT-100	92,97%	99,17%	99,78%	99,80%
Dutch	98,09%	99,69%	100,00%	100,00%
German	99,69%	99,69%	99,57%	96,76%
Chinese	99,14%	100,00%	100,00%	99,84%
Japanese	90,78%	84,30%	93,55%	95,16%

6. táblázat: Fix pont alapú újramintavételezés automatikus osztályozó pontosságára gyakorolt negatív hatásának arányai

Számos adatbázis-paraméter pár esetén 100%-ban rontott az átlagos hibaarányon ennek az újramintavételezési algoritmusnak az alkalmazása, vagyis minden egyes alkalommal, amikor sor került a használatára. A legjobb esetben is az összes konfigurációpár több, mint 74%-ban rontott az eredményeken.

Az egyértelmű eredmények okát azonban egészen könnyen meg lehet határozni. Az adatbázisok, melyeken a méréseket végeztük, mind egyenletes mintavételi frekvenciával rendelkeznek. Ebből kifolyólag, amennyiben a pontok száma alapján meghatározott időköz hossza nem egyezik az eredeti mintavételi időközökkel, akkor rengeteg rögzített pontot cserél az algoritmus interpolált pontokra. Ezzel az aláírások leírásának bizonytalanságát növelve.

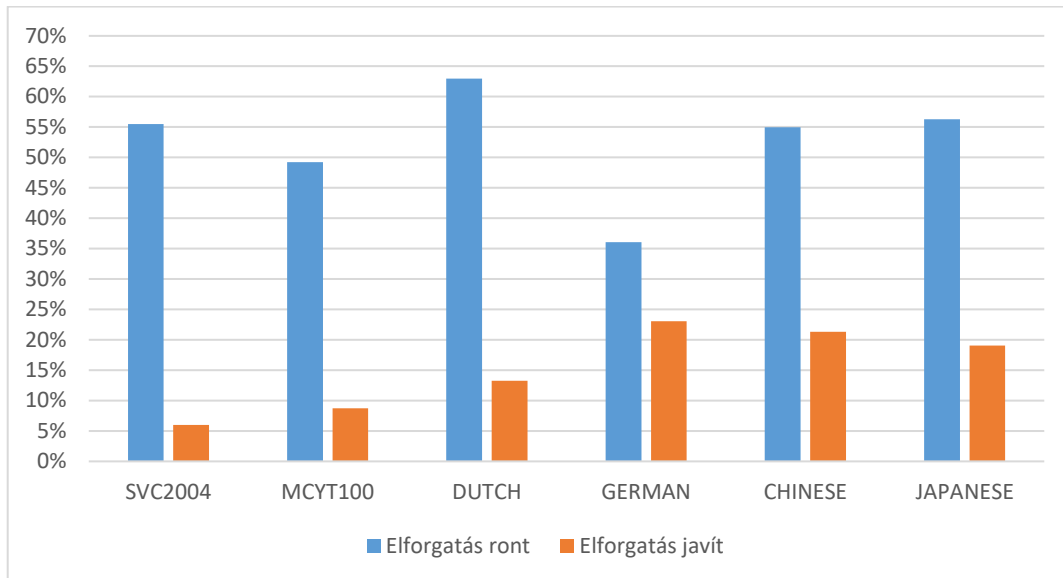
Az algoritmus megvalósításakor a másik fő gondolat az aláíráspontok számosságának különbségeinek megszüntetésére irányult. Ezzel azonban jelen rendszer esetében nem kell külön foglalkozni, hiszen a DTW algoritmus önmagában megoldást nyújt azzal, hogy definíció szerint képes kezelni az ilyen helyzeteket.

Az további eredmények torzulásának megakadályozása érdekében, innentől kezdve csak azoknak a mérési konfigurációknak az eredményeit vizsgálom, amik nem alkalmazzák ezt a megközelítést.

7.2 Elforgatás hatása

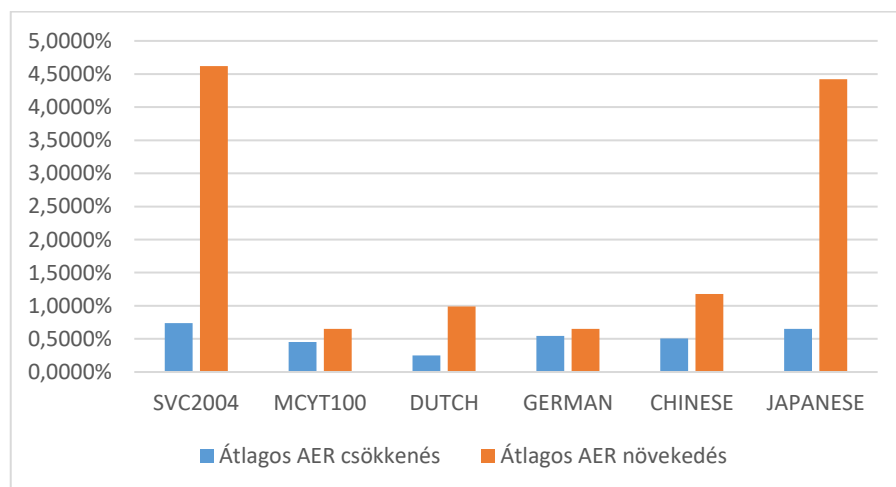
Az elforgatás szintén nem váltotta be a hozzá fűzött reményeket, ugyan eredményei kevésbé drasztikusak az előzőekhez képest. Hatásának értékelésére az előzőekben megismert konfigurációpárokat alkottam, és azt vizsgáltam, hogy ezek milyen százalékában ront és milyen százalékában javít az átlagos hibaarányon az elforgatás. Ezek arányát az optimális osztályozó esetében a 2. ábra szemlélteti. Ebből is

jól látszik, hogy az elforgatás alkalmazása is az esetek jelentősen nagyobb részében ront az eredményeken, mint amennyiben azokon javítani tud.



7. ábra: Az elforgatás javító és rontó eseteinek aránya az optimális osztályozót alkalmazó konfigurációi között

A pozitív és negatív kimenetelű esetek aránya mellett, meghatároztam azt is, hogy az elforgatásnak hála mennyivel tudták a javító esetek során az átlagos hibaarány értékét átlagosan csökkenteni, valamint, hogy a negatív hatás esetén átlagosan mennyivel növelték AER értékét. Ennek eredményeit a 8. ábra mutatja meg, ahonnan jól látható, hogy minden adatbázison nagyobb az a százalékpont, amennyivel átlagosan emelkedik az átlagos hibaarány az elforgatás negatív hatása következtében, mint az az érték, amivel pozitív hatás esetén csökkenteni tudja.



8. ábra: Elforgatás hatása az optimális osztályozó által elért AER értékek átlagos változását illetően

Nem elég tehát, hogy sokkal többször van negatív hatása az elforgatásnak, mint pozitív, ennek fejében ráadásul a káros hatásának mértéke is nagyobb a jótékonyhoz képest. Az automatikus osztályozó eredményei nagyon hasonlóan alakulnak, így azt nem fogom részletezni.

Az eredmények láttán a következő határozott megfigyelésem az, hogy általában ezt az elforgatásimplementációt sem érdemes alkalmazni az előfeldolgozás során. A világos eredmények ellenére az elforgatást nem zárom ki olyan mértékben, mint azt az előzőekben az újramintavételezési algoritmussal tettem, ugyanis a Dutch és a German adatbázisok esetében az összes alkalmazott konfiguráció közül a legjobb eredményt olyan adta, amelyik elforgatást is használt.

7.3 Távolságfüggvény megválasztása

A távolságfüggvény megválasztására a mérések során két lehetőség volt, mégpedig egy adott konfigurációban vagy Euklideszi-távolság alapján működik a DTW, vagy Manhattan-távolság alapján. Az eredmények átlagát tekintve első ránézésre akár jelentéktelennek is tűnhet, hogy melyiket választjuk, mert nagyon hasonló értékeket mutatnak. Az összes, az adott távolságfüggvény szerinti konfiguráció által elért átlagos hibaarányok átlagolása után azt kapjuk, hogy az Euklideszi-távolság 18,16%-os átlagos AER értéket ért el, míg a Manhattan-távolság 18,02%-ot. Hasonlón ezek adatbázisonkénti eredményeik nagyon közel vannak egymáshoz.

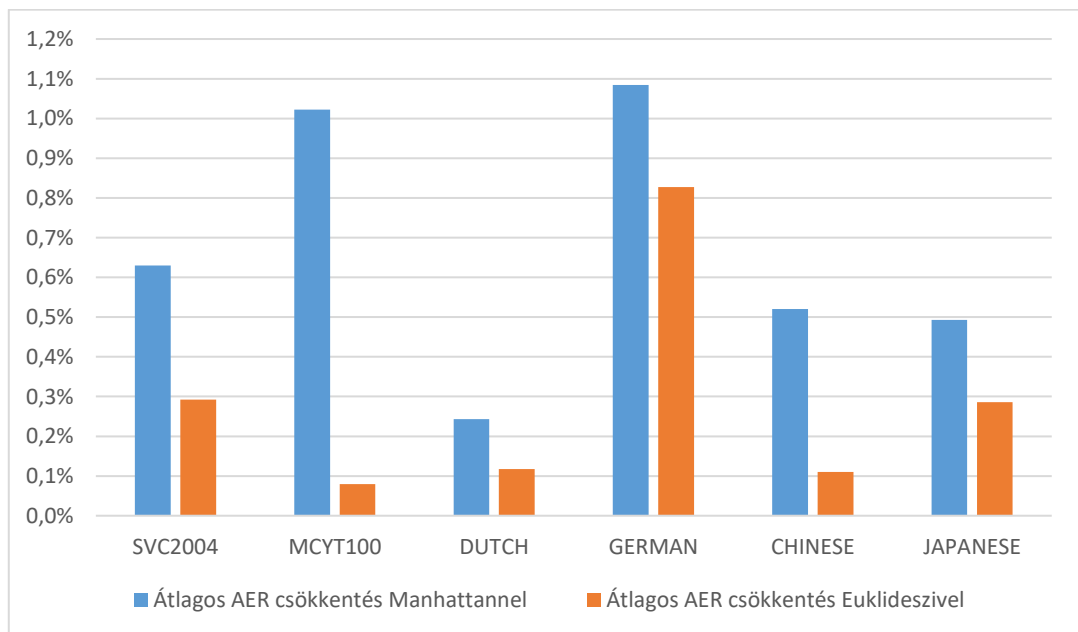
Az eredmények mélyebb analízise azonban egyértelművé tudta tenni, hogy melyik a jobb választás. A távolságfüggvények vizsgálata során is elvégeztem azon konfigurációk páronkénti összehasonlítását, melyek csak a távolságfüggvény szempontjából különböznek. Ennek eredménye a 7. táblázat, ami adatbázisokra és osztályozótípusokra bontva mutatja meg, hogy az esetek hány százalékában bizonyult eredményesebbnek az egyik vagy a másik távolságfüggvény, illetve, hogy milyen arányban értek el azonos átlagos hibaarányt.

Ez alapján jól látható, hogy az esetek túlnyomó részében valóban mindegy, hogy melyik távolságfüggvényt alkalmazzuk, mindkettővel ugyanolyan átlagos hibaarányt tudtunk elérni. Mindamellet az is tisztán kiolvasható, hogy azoknak az eseteknek a százalékos aránya egy tízes nagyságrenddel nagyobb, amikor a Manhattan-távolság hozott jobb eredményeket, mint azoknak, amikor az Euklideszi.

Adatbázis	Optimális osztályozó			Automatikus osztályozó		
	Manhattan	Euklideszi	Egyenlők	Manhattan	Euklideszi	Egyenlők
SVC2004	21,55%	2,54%	75,91%	21,94%	4,26%	73,80%
MCYT-100	29,86%	2,49%	67,65%	27,12%	6,98%	65,89%
Dutch	18,87%	4,96%	76,17%	21,09%	9,10%	69,80%
German	26,76%	6,02%	67,23%	23,83%	9,14%	67,03%
Chinese	23,71%	2,73%	73,55%	23,75%	5,70%	70,55%
Japanese	16,09%	4,30%	79,61%	19,22%	6,91%	73,87%

7. táblázat: Távolságfüggvények összehasonlítása az adott távolságfüggvénnyel jobb eredményt elérő esetek arányai alapján

Azon túlmenően, hogy a Manhattan-távolság alkalmazása az esetek nagyobb százalékában javít az AER értékén az Euklidesztávolság használatához képest, mint fordítva, az előbbi esetek által elért átlagos csökkentés az AER értékeken százalékpontokban kifejezve nagyobb, mint fordított esetben.



9. ábra: A távolságfüggvények átlagos hibaarány csökkentésének átlagos értéke optimális osztályozó használata mellett

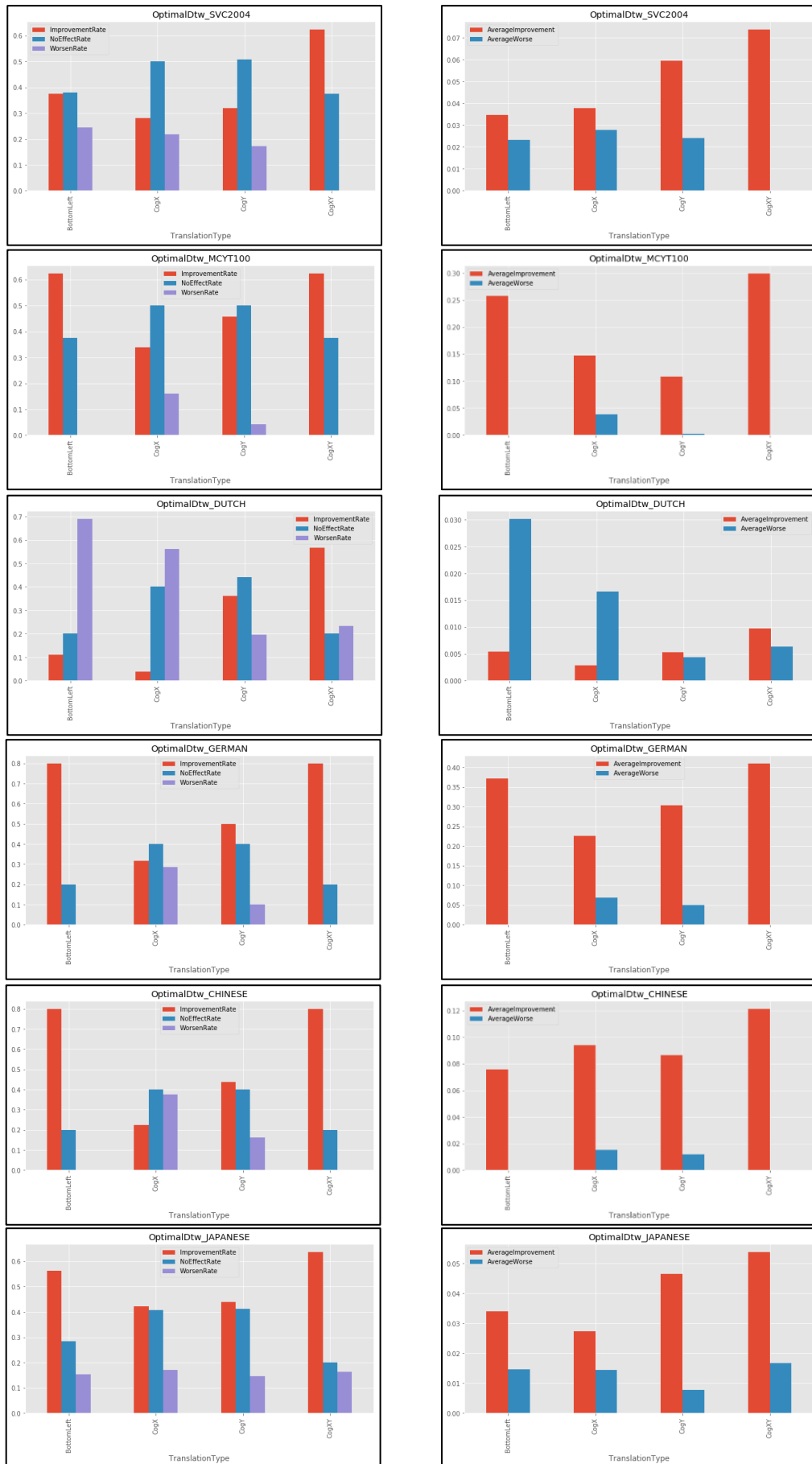
7.4 Eltolás hatásai

Az eltolási algoritmust a korábbiakban bemutatott négy különböző konkrét megvalósításra teszteltem. A fő célom ezzel az volt, hogy e négy megoldás közül meg tudjam határozni, hogy melyiket érdemes választani egy hitelesítő rendszer tervezése során.

A 10. ábra célja nem az eltolási megközelítések eredményességének mély elemzése. Segítségével egy átfogó képet szeretnék adni arról, hogy a négy eltolási megközelítés közül melyik segítette elő a hitelesítés pontosságának növelését a legnagyobb mértékben, vagyis melyiket érdemes választani.

Az ábrán látható diagrammátrix egyes sorai azt határozzák meg, hogy az adott diagramm eredményei melyik adatbázishoz tartoznak. Az adatbázisok egymást követve az eddig megszokott sorrendben kerülnek felsorolásra. A mátrix első oszlopának diagrammjai a korábbiakhoz hasonló módon a javító (piros), semleges (kék) és rontó (lila) esetek arányait szemlélteti. A második oszlopban az átlagosan elért AER csökkenést (piros) és növekedést (kék) láthatjuk. Emlékezzünk rá, hogy a hitelesítés pontosságának emelkedését az átlagos hibaarány (AER) értékének csökkenése jelenti. Minden diagram balról jobbra haladva BottomLeft, CogX, CogY, CogXY sorrendben ábrázolja a négy eltolási megközelítést. Az ábra az optimális osztályozó eredményeit tartalmazza, ám az automatikus osztályozóval elért eredmények is hasonló tendenciát mutatnak.

A 10. ábra gyors, ám figyelmes átfutása után kiolvasható, hogy a négy algoritmusimplementáció közül a CogXY, vagyis a két tengely menti súlypontba igazítás bizonyult a legeredményesebbnek. Ez az adatbázisok többségén egyetlen benchmark konfiguráció esetén sem okozott eredményromlást, ám a pozitív hatású esetek arányát tekintve minden adatbázisnál az élmezőnyben járt. Ugyanígy az átlagos hibaarány értékének átlagos csökkentése tekintetében is minden adatbázison futtatva élen járt.



10. ábra: Eltolási megoldások összehasonlítása azok átlagos hibaarány értékére gyakorolt pozitív, negatív és semleges hatásának arányai, valamint az átlagos AER módosító hatásuk alapján

7.5 Skálázás hatásai

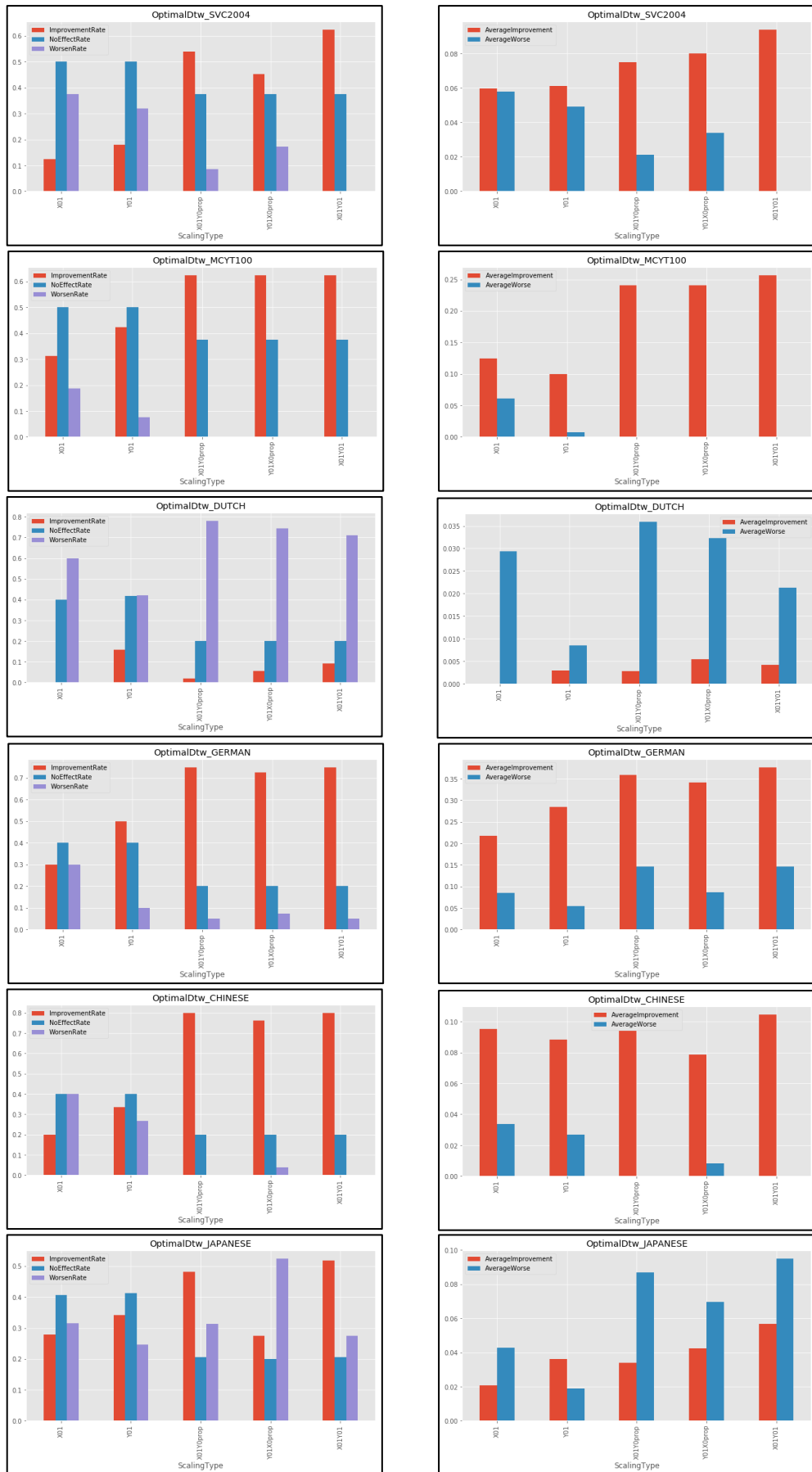
A skálázás megvalósítása során öt konkrét megoldást vezettem be, melyek közt van egyszerű és aránytartó skálázás egyaránt és amelyeken a mérések is történtek. A céloom ebben az esetben is az eltolásnál látottakhoz hasonlóan ezen öt implementációnak az összehasonlítása volt.

A 11. ábra bemutatásának célját és felépítését tekintve teljes mértékben megegyezik 10. ábra tulajdonságaival. Ez az ábra is az optimális osztályozó eredményeit tartalmazza. Egyedüli különbség, hogy a 11. ábra a skálázási megoldásokat mutatja be, ennek megfelelően balról jobbra haladva X01, Y01, X01Y0prop, Y01X0prop, X01Y01 megközelítések eredményei láthatók a kis diagrammokon.

Az eredményekből egyértelműen látszik, hogy a Dutch adatbázison semmilyen formában nincs értelme az aláírásokat skálázni, ám az előbbi fejezetben láthattuk, hogy az eltolás alkalmazása célszerű lehet. A többi adatbázis tekintetében megfigyelhetjük, hogy határozottan ajánlott mind a két koordinátatulajdonság mentén elvégezni a skálázás, hiszen ezzel az esetek nagyobb arányában, és láthatóan nagyobb mértékben lehet javítani a hitelesítési eredményeken, mintsem csak az egyik tulajdonság felhasználásával.

Ugyanitt fontos megemlíteni azt is, hogy az egyszerű, két független skálázást végrehajtó X01Y01 megközelítés hasonlóan jó vagy jobb eredményeket ért el a diagrammátrix mindkét oszlopa alapján, mint a bonyolultabb aránytartó skálázási megoldások.

Mіндеzt összegezve kijelenthetjük, hogy az eredmények alapján határozottan az X01Y01 elnevezésű, két tulajdonság alapján működő, egyszerű skálázást érdemes választani.



11. ábra: Skálázási megoldások összehasonlítása azok átlagos hibaarány értékére gyakorolt pozitív, negatív és semleges hatásának arányai, valamint az átlagos AER módosító hatásuk alapján

7.6 Eredmények értékelése

Az elvégzett munkámat alapvetően sikeresnek tekintem a mérési eredmények alapján. Az eredeti célomnak megfelelően, sikerült olyan megfigyeléseket megfogalmaznom, amiket széleskörű méréssorozat eredményeivel alá is tudok támasztani, és a továbbiakban hasznosak lehetnek, nagy segítséget nyújthatnak DTW alapú online aláíráshitelesítő rendszerek megtervezése során. A távolságfüggvények, eltolási és skálázási algoritmusok tekintetében több alternatívát fogalmaztam meg, melyek közül a legjobb választást mindhárom esetben sikerült meghatároznom a mérési eredmények alapján.

Míndeközben olyan automatikus osztályozót használó konfigurációk kerültek meghatározásra, melyekkel az irodalommal teljes mértékben összhangban lévő hitelesítési pontosságot sikerült elérni.

8 Összegzés

Jelen dolgozatban különböző előfeldolgozó lépéseket mutattam be, melyeket implementáltam is, majd egy DTW alapú online aláíráshitelesítő környezetben felhasználva átfogó méréssorozatot végeztem a hitelesítési eredményekre gyakorolt hatásaik vizsgálata céljából.

A dolgozatot a tématerület bemutatásával és annak jelentőségének meghatározásával kezdtem. Ezután ismertettem a munkám szempontjából releváns szakirodalmat, a terület mai állását és a leggyakrabban használt megközelítéseket. Mivel régebb óta foglalkozom ezzel a témával, így a szakirodalom bemutatása után a tavalyi TDK dolgozatom releváns részeinek összegzése következett.

Ezután a legfrissebb munkám tételes leírása következett. Ennek során számos előfeldolgozási algoritmust definiáltam és ismertettem azt, hogyan valósítottam meg a gyakorlatban az ezek mögött álló elméleti megfontolásokat. Ezután bemutattam, hogy az algoritmusaim tesztelésére milyen széleskörű méréssorozat került megvalósításra. Végezetül ismertettem a mérési eredmények alapján levonható legerősebb megfigyeléseimet, és levontam azt a következtetést, hogy az általam elért eredmények az irodalommal összhangban vannak.

Mindeközben sikerült 2%-os átlagos hibaaány alatti eredménnyel rendelkező konfigurációt meghatároznom, mely a gyakorlatban is használható automatikus osztályozó használatára épül.

9 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*, pp. 2276-2297, 2018.
- [2] 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.
- [3] C. E. v. d. H. K. Y. F. a. L. G. V. V. L. Blankers, „ICDAR 2009 Signature Verification Competition,” in *IEEE*, Barcelona, Spain, 2009.
- [4] 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.
- [5] M. I. Malik, M. Liwicki, L. Alewijnse, 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.
- [6] 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 (SigWiComp2015),” in *IEEE*, Tunis, Tunisia, 2015.
- [7] F. D. G. S. D. C. Anil K. Jain, „On-line signature verification,” *Pattern Recognition*, %1. kötet35, %1. szám12, pp. 2963-2972, 2002.
- [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] Chadha, Aman, Divya Jyoti és M. Mani Roja, „Rotation, Scaling and Translation Analysis of Biometric Signature Templates,” *arXiv preprint arXiv*, %1. kötet2, pp. 1419-1425, 2011.
- [11] Nilchiyan, Yusof és Alavi, „Statistical on-line signature verification using rotation-invariant dynamic descriptors,” *Control Conference (ASCC)*, pp. 1-6, 2015.
- [12] Khan, Khalid, Khan és Ahmad, „On-line signature verification by exploiting inter-feature dependencies,” *Pattern Recognition*, %1. kötet2, pp. 796-799, 2006.
- [13] 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.
- [14] 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.
- [15] 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.
- [16] 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.
- [17] 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.
- [18] 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.

- [19] P. Senin, „Dynamic Time Warping Algorithm Review,” Honolulu USA, December 2008.
- [20] Alisher Kholmatov és Berrin Yanikoglu, „Identity authentication using improved online signature verification method,” *Pattern Recognition Letters*, %1. kötet26, %1. szám15, pp. 2400-2408, 2005.
- [21] M. Faundez-Zanuy, „On-line signature recognition based on VQ-DTW,” *Pattern Recognition*, %1. kötet40, %1. szám3, pp. 981-992, 2007.
- [22] 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.
- [23] 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.
- [24] M. Okawa, „Template Matching Using Time-Series Averaging and DTW With Dependent Warping for Online Signature Verification,” *IEEE Access*, %1. kötet7, pp. 81010-81019, 2019.
- [25] Y. Jia, L. Huang és H. Chen, „A Two-Stage Method for Online Signature Verification Using Shape Contexts and Function Features,” *Sensors*, %1. kötet19, %1. szám8, p. 1808, 2019..
- [26] LorisNanni, „Experimental comparison of one-class classifiers for online signature verification,” *Neurocomputing*, %1. kötet69, %1. szám7-9, pp. 869-873, 2006.
- [27] C. Sekhar V., P. Mukherjee, D. S. Guru és V. Pulabaigari, „Online Signature Verification Based on Writer Specific Feature Selection and Fuzzy Similarity Measure,” in *Applications of Computer Vision and Pattern Recognition to Media Forensics, CVPRW*, Long Beach, California, 2019.
- [28] D. I. Dikii és V. D. Artemeva, „Online Handwritten Signature Verification System Based on Neural Network Classification,” in *2019 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*, Saint Petersburg and Moscow, Russia, 2019.

- [29] C.-Y. Park, H.-G. Kim és H.-J. Choi, „Robust Online Signature Verification Using Long-term Recurrent Convolutional Network,” in *2019 IEEE International Conference on Consumer Electronics (ICCE)*, Las Vegas, NV, USA, 2019.
- [30] C. Sekhar, P. Mukherjee, D. S. Guru és V. Pulabaigari, „OSVNet: Convolutional Siamese Network for Writer Independent Online Signature Verification,” in *International Conference on Document Analysis and Recognition (ICDAR 2019)*, Sydney, Australia, 2019.
- [31] Yu Qiao, Xingxing Wang és Chunjing Xu, „Learning Mahalanobis Distance for DTW based Online Signature Verification,” in *IEEE*, Shenzhen, China, 2011.
- [32] Ortega-Garcia, Fierrez-Aguilar, Simon, Gonzalez, Faundez-Zanuy, Espinosa és Escudero, „MCYT baseline corpus: a bimodal biometric database,” in *IEE Proceedings-Vision, Image and Signal Processing*, 2003.
- [33] C. L. Szücs, „On-line aláírás-hitelesítési algoritmusok vizsgálata,” <http://tdk.bme.hu/VIK/information/Online-alairashitelesitesi-algoritmusok>, 2018..
- [34] „Signature verification library,” [Online]. Available: <https://github.com/sigstat/sigstat>.
- [35] „BME Szuperszámítógép,” [Online]. Available: <https://superman.eik.bme.hu/>.
- [36] 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.
- [37] 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.
- [38] 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.
- [39] E. Griechisch, „ONLINE SIGNATURE VERIFICATION AND HANDWRITING CLASSIFICATION,” 2018.

- [40] 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.