MNB-BME Együttműködés
2020/2021
Digitalizáció, mesterséges
intelligencia és adatkorszak Műhely

Kiséry Máté Soma, Fábián István,
Gulyás Gábor György

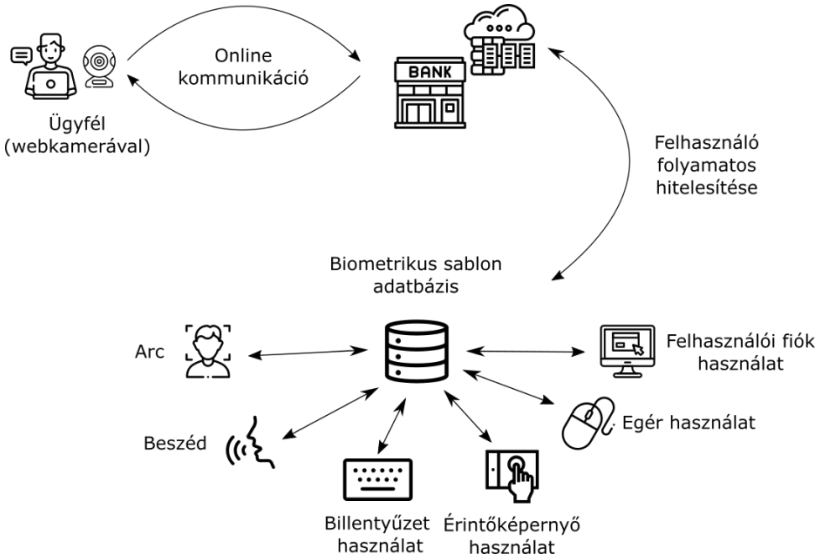
ARCFELISMERÉS ADATVÉDELEMMEL

BUDAPEST, 2021

Tartalomjegyzék

Vezetői összefoglaló

A biometrikus azonosító rendszerek valamilyen egyedi biológiai tulajdonságot használnak fel a különböző személyek azonosítására (ujjlenyomat, arclenyomat, járás- vagy beszéd- stílus, szem vagy írisz mintázat stb.). E tanulmány, különös hangsúlyt fektetve az arcfelismerésre, bemutatja az egyre szélesebb körben elterjedő biometrikus rendszerek csoportosítását, működését és főbb jellemzőit, illetve példát ad lehetséges alkalmazásukra a banki szektorban (1. ábra).

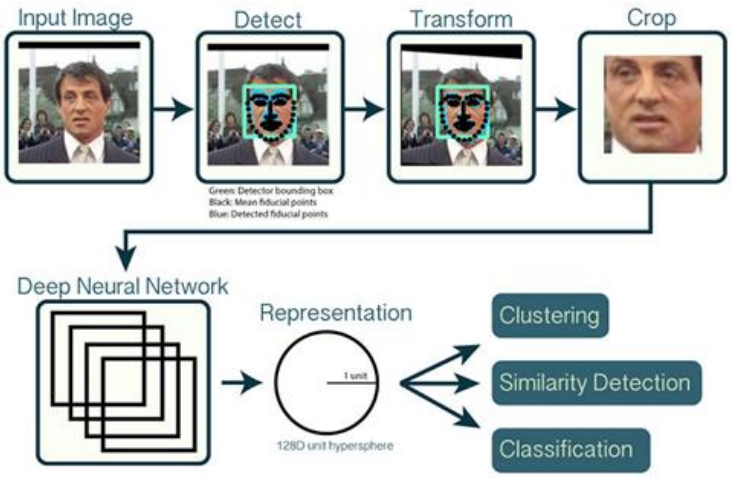


1. ábra – Biometrikus azonosítás alkalmazása a bank szektorban, az online bankolás biztonságosabbá tételére. Az online kommunikáció során az ügyfél folyamatos biometrikus hitelesítése zajlik a háttérben, így megakadályozva internetes csalásokat.

A biometrikus rendszerek alkalmazásával, beleértve az arcfelismerést is, számos előny és hátrány járhat. A biometrikus adatok kezelése ugyanis adatvédelmi kockázatot jelent, s éppen ezért kezelésüket az európai általános adatvédelmi rendelet (GDPR) is szabályozza. Ez szükségessé teszi az adatvédelmi kockázatokat csökkentő technikák kidolgozását és alkalmazását. Ilyen módszer például a bemutatásra kerülő két biometrikus sablon védelmi eljárás, melyek az arcfelismerés adatvédelmi kockázatait jelentős mértékben csökkentik (az egyik hashelésre, a másik titkosításra építve).

A terület tanulmányozásához először szükséges a különböző arcfelismerési eljárások működésének megértése, beleértve az arcdetektálás és az arcfelismerés részleteit is. A mély tanulásra épülő arcfelismerési rendszerek sematikus működése a 2.

ábrán látható. A neurális háló a bementén egy arcot ábrázoló képet kap, melyen az arc pontos koordinátáit az arcdetektáló algoritmus határozza előzőleg meg, majd a kimenetén egy az arc jellemzőit leíró ún. arclenyomat vektort (angolul face embedding) állít elő.



2. ábra – Az arcfelismerés folyamata. A bemeneti képen meghatározzuk az arc pozícióját és orientációját, majd amennyiben szükséges, különböző transzformációkkal frontális képpé alakítjuk. A csak az arcot tartalmazó képből egy neurális háló létrehozza az arcot leíró struktúrát, amelyet különböző módszerekkel (pl. osztályozó algoritmusokkal) vizsgálhatunk arcfelismerés szempontjából. (ábra forrása: [13])

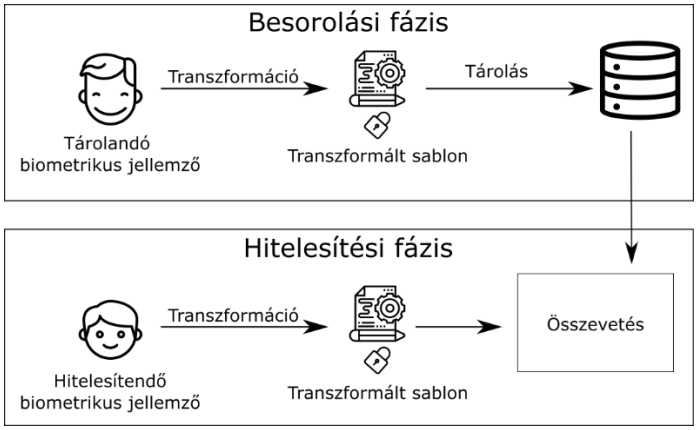
A ma létező legkorszerűbb, mély tanulásra épülő arcfelismerési programozási könyvárak között vannak privát (pl. a Facebook által fejlesztett DeepFace, illetve a Google által fejlesztett FaceNet) és szabadon hozzáférhető, open source (pl. OpenCV, dlib, InsightFace) megoldások is. Minden könyvtárhoz különböző arcdetektor, illetve különböző architektúrájú neurális háló tartozik, s az egyes könyvtárak által előállított arclenyomat vektorok hosszúságában is lehet különbség. Ezen felül lehet különbség az egyes könyvtárak arcdetektálási pontosságában (hamis pozitív és hamis negatív találatok) és futási idejében is, így a megfelelő könyvtár kiválasztása mindig feladat specifikus.

Az arcfelismerés során keletkező arclenyomat vektorok azonosításra történő felhasználására több különböző lehetőség is van. Az egyik a klaszterezés (csoportosítás), mely egy hasonlósági metrikán alapuló küszöb sémát alkalmaz, egy másik lehetőség pedig a gépi tanulásra épülő osztályozás, melyet szintén lehet hitelesítésre alkalmazni. Több különböző gépi tanulási osztályozási technika is létezik, mint például a döntési fa (decision tree), véletlen erdő (random forest), support vector machine, illetve a neurális

háló, melyek mind képesek különböző személyek arclenyomat vektoraik alapján történő osztályozására, felismerésére.

Az arcfelismerés gyakorlati alkalmazásának adatvédelmi nehézségei és kockázatai is vannak. A mély tanulásos rendszerekbe a tanító adathalmazokon keresztül bekerülő torzítások által jelentett kockázatok jelentősek lehetnek, például afroamerikai és ázsiai rasszba tartozó emberek esetén előfordul a magasabb hamis felismerési arány. A tömeges arcfelismerés által jelentett társadalmi kockázatok is kiemelkedő jelentőségűek, ugyanis ez a technológia sosem látott tömeges megfigyelést tehet lehetővé helytelen alkalmazás esetén. Továbbá az arclenyomatok magánszférában történő feldolgozásával kapcsolatos adatvédelmi probléma, hogy bár nyílt keresési halmaz esetén az arclenyomatok nem nyújtanak közvetlen azonosításra hatékony lehetőséget (pl. nehéz ez alapján végigkeresni egy közösségi hálót), de mivel azokból egész pontosan ki lehet nyerni az alapvető demográfiai adatokat (pl. nem, kor, rassz), így a közvetett keresés hatékonnyá válhat, ami pedig jelentős adatvédelmi kockázatként jelentkezik.

Mivel az emberek biológiai tulajdonságai (pl. arc, ujjlenyomat) nehezen, vagy egyáltalán nem megváltoztathatóak vagy visszavonhatóak (mint jelszavak vagy PIN kódok tárolása esetén erre lehetőség van), így szükség van olyan eljárásokra, melyekkel ezek a kockázatok mérsékelhetők. Ezeket az eljárásokat nevezzük biometrikus sablonvédelmi eljárásoknak (3. ábra). A biometrikus sablonvédelemnek különböző alkalmazási lehetőségei vannak, melyek segítségével adatvédelem barát módon lehet arcfelismerést megvalósítani (de a tárgyalt módszerek többnyire alkalmasak más biometrikus azonosítási eljárásokhoz is).



3. ábra – A biometrikus sablonvédelem működésének sematikus bemutatása. A besorolási fázis során a hitelesítendő személyről elmentünk egy referencia biometrikus sablont, s hitelesítés során

ehhez viszonyítunk. Az eredeti biometrikus jellemzők tárolása helyett azokat transzformáljuk, s csak a transzformált sablonok kerülnek tárolásra, így csökkentve az adatvédelmi kockázatot.

Az egyik biometrikus sablonvédelmi eljárási lehetőség a hashelés egy különleges változatára, a helyérzékeny hashelésre (angolul locality sensitive hashing) épülő technika, mely egy ún. véletlen projekció nevű eljárást használ az arclenyomat vektorok módosítására. Így olyan biometrikus sablonokat hoz létre, melyek a távolságokat tartó transzformáció segítségével egy másik, az eredetitől eltérő térbe kerülnek. A módosított lenyomatok azonosításra továbbra is alkalmazhatók, viszont az egyirányú, veszteséges leképzés miatt vissza nem követhetők, és ezért csökken az adatvédelmi kockázat.

A másik biometrikus sablonvédelmi technika a homomorfikus titkosításra épít, amely lehetővé teszi közvetlenül a titkosított adatokon történő műveletek végzését. Esettanulmányi példának bemutatunk egy háromszereplős adatvédelem barát arcfelismerési rendszert, melyben az egyes szereplők egy kamera, egy külső szerver, illetve egy helyi szerver. A rendszer lényege, hogy a kamera az általa készített képekből kivonja az arclenyomatokat, majd ezeket titkosítva küldi tovább a külső szervernek, amely a homomorfikus titkosítás miatt képes összevetni őket az egyes emberekről szintén titkosítva tárolt referencia arclenyomatokkal. Az összevetés eredményét a helyi szerver fogadja, amely a titkosítást feloldva megtudja, hogy mikor melyik személy halad el a kamera előtt. Így olyan rendszer tervezhető, melyben nincs szükség az érzékeny arclenyomatok tárolására, s egyik szereplő sem fér hozzá titkosítatlanul tárolt arclenyomatokhoz.

1 Bevezetés

A távbankolás és az interneten keresztüli digitális fizetések egyre elterjedtebbek világszerte, s ezt a terjedést a COVID-19 járvány még jelentősebben felgyorsította. Magyarországon a Gránit Bank volt az első, mely videóbankolást ajánlott ügyfeleinek akár számlanyitást is lehetővé téve, elektronikus ügyfélazonosítás alkalmazásával.

Digitális kommunikáció során kiemelten fontos a személyek megfelelő azonosítása és hitelesítése, hiszen egy rosszindulatú harmadik fél is megpróbálhatja hamisan a bank egyik ügyfelének kiadni magát, melyben segítségére lehetnek kártékony szoftverek vagy különböző pszichológiai manipulációs technikák.

Emlékezetes példa lehet az az eset, amikor hang manipulációval csaltak ki 220 ezer eurót egy német tulajdonú angol energetikai cég vezérigazgatójától, aki azt hitte, hogy a német cég vezetőjével beszél, s az ő utasításai szerint cselekszik, miközben a vonal túlsó végén valójában egy mesterséges intelligenciával előállított deepfake hang utánozta főnöke hangját [44]. Hasonló veszélyek már ma is reális visszaélések lehetnek videóbankolás során. Ugyan a deepfake technika még nem tart ott, hogy teljesen élethűen leutánozható legyen valaki egy videóhívás során, de a technológia rohamos fejlődése miatt néhány éven belül ez komoly veszélyt jelenthet a banki szektorra is nézve.

Az ezekhez hasonló veszélyek kiküszöbölésére számos elsődleges (pl. PIN kód) és másodlagos (pl. viselkedés egyezőség vizsgálata) hitelesítési megoldás alkalmazható. Például videóhívás során a háttérben arcfelismerési technológia segítségével lehetőség van arra, hogy az ügyfélről tárolt arcképes igazolványon (például személyi igazolvány) található képet összevessük a videóhívásban lévő arccal, s amennyiben nincs egyezés, a rendszer riaszthasson. Továbbá lehetséges a videóhívás alatt is folyamatosan figyelemmel tartani az ügyfél viselkedését és azt összevetni a szokásos viselkedésével (pl. egér mozgás, touchpad használat, billentyűzet használat, webes felhasználói felület használata), s gyanús eltérés észlelése esetén szintén riasztást adni.

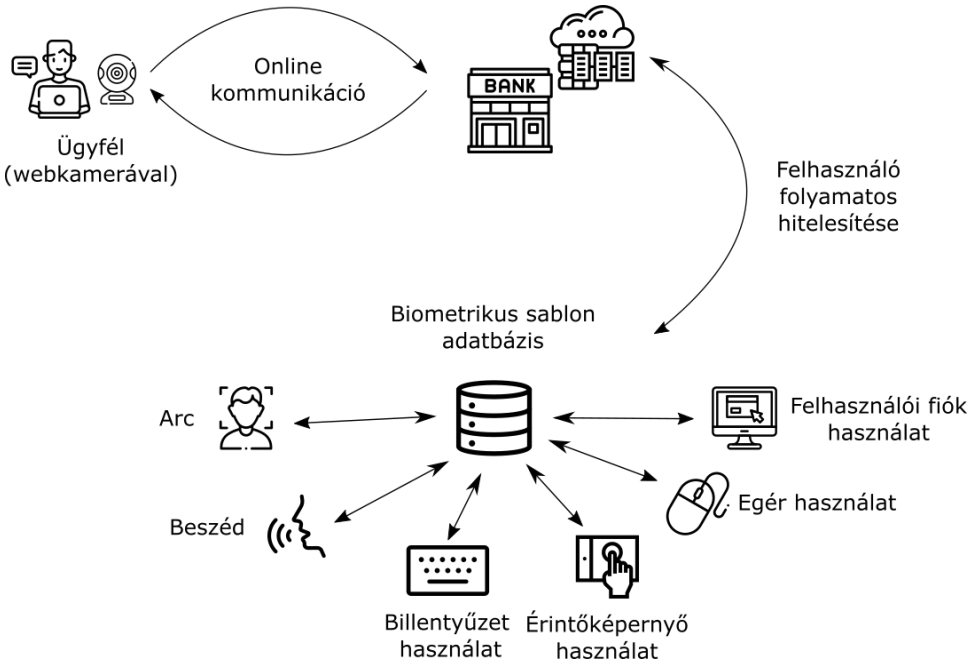
A távbankolás során történő hitelesítés egyik lehetséges megoldása a biometrikus azonosítási eljárások alkalmazása. A biometrikus azonosítók bizonyos biológiai tulajdonságok azonosítás céljából történő feldolgozása során állnak elő, s alkalmasak arra, hogy megfelelő technológiával különböző emberek gyorsan felismerhetők és

azonosíthatók legyenek. Ilyen biometrikus azonosítókat lehet előállítani például az ujjlenyomat, szem szivárványhártya mintázat, arckép, hang stb. alapján.

A biztonság növelése érdekében a biometrikus azonosítási eljárások általában nem önmagukban, hanem valamilyen hagyományos azonosítóval (például jelszó vagy PIN kód) együtt kerülnek alkalmazásra, mint másodlagos hitelesítési faktorok. Ennek következtében a bankoknak nem csak az ügyfél személyes megjelenése esetén kell érzékeny adatokat kezelniük (például amikor egy ügyfél személyesen megjelenik egy bankfiókban és erről egy kamera digitális felvételeket készít), hanem távbankolás során biometrikus azonosítás céljából is szükségessé válhat például az ügyfél arcáról készült videófolyamok kezelése.

Ez azonban jelentős adatvédelmi kockázatokkal jár. Az Európai Unió Általános adatvédelmi rendelete (GDPR) is szabályozza a biometrikus adatok kezelését, sőt, kezelésük során a különleges adatokra vonatkozó szabályokat is be kell tartani [1]. Például arcfelismerés technológia alkalmazása során a felemerülő kockázatok közé tartozik, hogy az arcképekről készült azonosítási célból készített arclenyomat vektorok (a biometrikus azonosítók) rosszindulatú harmadik fél kezébe jutása esetén demográfiai adatokat szivárogtathatnak ki az arclenyomathoz tartozó személyekről [12], vagy az arclenyomat alapján a harmadik fél képes lehet akár az eredeti arc visszaállítására is [8], mely lehetőséget adhat a fentebb említett deepfake visszaélésekre. A kockázatok csökkentése érdekében tehát a biometrikus adatok feldolgozása során megfelelő kriptográfiai titkosítást kell alkalmazni, illetve gondoskodni kell a hozzáférés-ellenőrzésről is.

Jelen tanulmány a biometrikus azonosítás, azon belül is az arcfelismerés alkalmazásával kapcsolatos technológiai eljárásokat és a kapcsolódó kockázatokat mutatja be, illetve azt, hogy hogyan lehet ezeket az adatokat biztonságosan kezelni és a kockázatokat csökkenteni.



Ábra 4 - Online bankolás biztonságosabbá tétele másodlagos biometrikus azonosítással

2 A biometrikus azonosításról

Azonosítás alatt azt értjük, amikor egy rendszer vagy alkalmazás felhasználójának személyazonosságát meghatározzuk több lehetséges felhasználó közül. Ezzel szemben a hitelesítés azt jelenti, hogy egy felhasználó hitelt érdemlően bebizonyítja, hogy ő valóban az, akinek állítja magát. Máshogy megfogalmazva az azonosítás azt a kérdést válaszolja meg, hogy *"ki vagy te?"*, míg a hitelesítés arra a kérdésre ad választ, hogy *"valóban te vagy XY?"*.

Az emberiséget régóta foglalkoztatja az azonosítás és hitelesítés gondolata. Például az ókori kínai és babiloni hivatalnokok is használták már az ujjlenyomataikat hivatalos dokumentumok hitelesítésére, sőt, egyes leletek szerint mindkét népnél bevett szokás volt bűnözők ujjlenyomatát rögzíteni, s Európában is már a 17. századtól foglalkoztak akadémikusok ujjlenyomat tanulmányozással [28]. Persze nem csak az ujjlenyomat volt használatban korábban. A 19. században Alphonse Bertillon, egy francia rendőr, egy olyan rendszert dolgozott ki visszaeső bűnözők azonosítására, mely az emberi test bizonyos anatómiai tulajdonságainak (fejszélesség, karhossz, magasság, tetoválások, anyajegyek stb.) rögzítésén alapult [29]. Mi több, a második világháború idején a távíró operátorok képesek voltak egymás felismerésére a Morse kódban használt jel és szünet sorozatok küldésének dinamikája alapján [2][30].

A fentebb felsorolt példák mind szorosan kapcsolódnak a biometrikus azonosítás és hitelesítés témakörébe. Az emberek azonosítása alapvetően háromféleképpen történhet: valami alapján, amit birtokolnak (pl. kulcs, dokumentum, kártya, jelvény), valami alapján, amit tudnak (pl. felhasználónév, jelszó) vagy valami alapján, ami rájuk jellemző (pl. ujjlenyomat, DNS, arc, hang, viselkedés). Ezek közül az utolsó, az emberek egyedi biológiai és viselkedésbeli tulajdonságainak felhasználása az, amivel a biometrikus azonosítás foglalkozik.

2.1 A biometrikus azonosítás működése

Biometrikus azonosítás és hitelesítés során egy adott ember valamilyen tulajdonságát, jellemvonását vetjük össze egy eltárolt biometrikus sablonnal, hogy eldöntsük, hogy a vizsgált ember megegyezik-e azzal, akiről a biometrikus sablon készült. Biometrikus

azonosítás történhet fiziológiai vagy viselkedési alapon [2]. A fiziológiai tulajdonságokat csoportosíthatjuk morfológiai és biológiai kategóriákra. A morfológiai kategóriába tartozik például az ujjlenyomat, a kéz alakja, az erezet mintázat, az írisz mintázat, az arc stb., míg a biológiai csoportba tartozik például a DNS, vér, nyál, vizelet és hasonlók alapján történő azonosítás. Viselkedés alapú azonosításra meg használható a beszédstílus (hang), gesztusok, testtartás, aláírás dinamika, billentyűzet használat, egérmozgatás, érintő képernyő használat, felhasználói fiók használat, böngészési előzmények stb. A különböző biometrikus azonosítási eljárásokhoz természetesen különböző megbízhatósági szint tartozik. Például fiziológiai, morfológiai azonosítók általában kevésbé változnak az idő múlásával vagy stresszhelyzetben, mint a viselkedés alapú azonosítók.

Ábra 5 - Biometrikus azonosítók csoportosítása [2]

A biometrikus rendszerek terjedése napjainkban egyre szélesebb körű, köszönhetően a társadalmi elfogadottság és az elérhető pontosság növekedésének, illetve a csökkenő szenzor áraknak és új szoftverek megjelenésének. Biometrikus rendszerekkel találkozhatunk többek között a kereskedelemben, bűnüldözésben, az egészségügyben, a határvédelemben, a bankszektorban és számos egyéb ágazatban is. Ez a tanulmány az arcfelismerés technológiára fókuszál, melynek szintén számos felhasználási területe van, mind a magán, mind az állami szektorban, melyek a későbbiekben lesznek részletesebben bemutatva

A biometrikus rendszerek terjedése számos előnyüknek köszönhető. Egyrészt hagyományos eljárásokkal szemben (pl. jelszó, kulcs) a biometrikus azonosítókat nem lehet elfelejteni vagy meghamisítani, bár bizonyos kockázatok felmerülnek, melyek az alábbiakban részletesebben ki vannak fejtve. Továbbá egyetlen biometrikus azonosító helyett lehet többet is vizsgálni, ez által növelve a biztonságot és a pontosságot (ezeket nevezzük multimodális biometrikus rendszereknek). További előny például az arcfelismerő rendszereknél, hogy mégcsak fizikai kontaktust sem igényelnek, mint például az ujjlenyomat vétel, ez által még könnyebb azonosítást téve lehetővé.

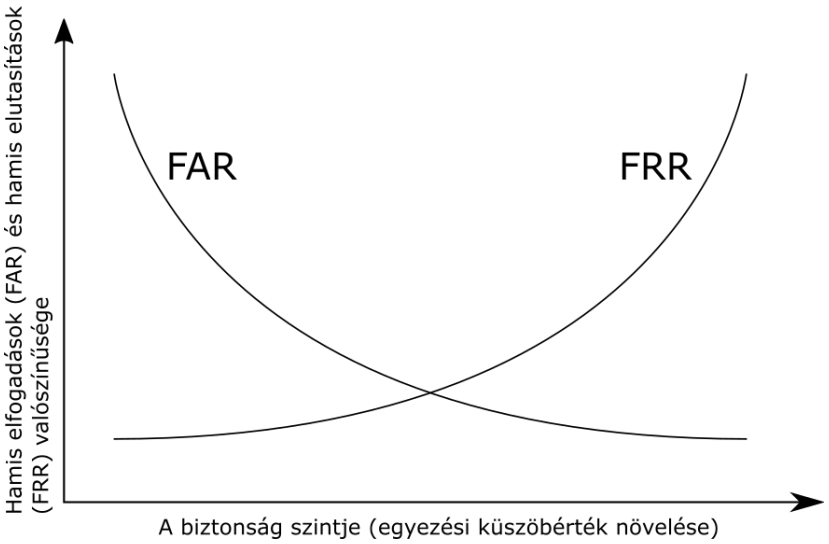
A banki világban is számos előnyös alkalmazásuknak köszönhetően vannak terjedőben a biometrikus rendszerek. Nem csak a banki ügyfeleknek jelent könnyebbséget, ha nem kell újabb jelszavakat és felhasználó neveket megjegyezniük, hanem például a csalás detektálásban és megelőzésben is segíthet a biometrikus azonosítók alkalmazása. Ahogy a bevezetőben is említésre került, lehetőség van például minden banki ügyfélről egyedi profil létrehozására, figyelembe véve mind morfológiai, mind viselkedésbeli biometrikus azonosítókat. Ezt követően gépi tanulás által felügyelt folyamattal lehetne azokat az eseteket kiszűrni, amikor egy csaló hamisan ki akarja adni magát a bank egyik ügyfelének. A profilok létrehozása során figyelembe vehető azonosítók például a billentyűzet használat, egér használat, felhasználói fiók használati szokások, illetve akár eszköz jellemzők is, mint például IP cím, MAC cím vagy geográfiai hely.

Ugyanakkor a biometrikus rendszerek sem járnak teljesen kockázat nélkül, illetve minden biometrikus azonosítási eljárásnak vannak előnyei és hátrányai a többivel szemben (erre mutat néhány példát a Függelékben található Táblázat 2. táblázat a teljesség igénye nélkül). Az egyik legvitatottabb kérdés például a biometrikus adatok védelme. Mivel azonosítás és hitelesítés során az új biometrikus mintákat az ügyfelekről tárolt biometrikus sablonokkal kell összevetni, ezért szükségszerű az ügyfelek biometrikus sablonjainak eltárolása központi adatbázisban, mely komoly adatvédelmi kockázatokat rejt. A központi adatbázisban való tárolás azért különösen veszélyes biometrikus azonosítók esetén, mert egy ember biológiai tulajdonságai általában nem megváltoztathatóak, illetve különböző szolgáltatók esetén is tárolásra kerülhetnek ugyanazok az azonosítók, így az adatok kiszivárgása komoly gondokat okozhat.

Ahogy a bevezetőben említettük, az ilyen problémák miatt a GDPR szerint is szigorú szabályozások vonatkoznak a biometrikus adatok kezelésére. A 9. cikk 1. bekezdése alapján a biometrikus adatok kezelése néhány kivételes helyzettől eltekintve tilos. További, ilyen adatok például a faji vagy etnikai származásra, politikai véleményre, vallási meggyőződésre, szakszervezeti tagságra, egészségügyi vagy szexuális irányultságra vonatkozó személyes adatok. A 9. cikk 2. bekezdése alapján kivételes helyzetnek minősül és megengedett ezen adatok kezelése, ha az érintett kifejezett hozzájárulását adta, az adatkezelés az érintettnek a konkrét jogait vagy érdekeit védi, ha az érintett fizikai vagy jogi cselekvésképtelensége folytán nem képes hozzájárulását adni, vagy ha az adatkezelés kifejezetten nyilvánosságra hozott adatokat érint vagy jelentős közérdek miatt szükséges. Továbbá a 17. cikk értelmében (törléshez való jog vagy „az elfeledtetéshez való jog") az érintettek jogosultak arra, hogy kérésükre az adatkezelő törölje a rájuk vonatkozó személyes adatokat, amennyiben az adatkezelés jogalapját az érintettek hozzájárulása képezte. A szabályok be nem tartása a GDPR értelmében nagy összegű pénzbírságokat vonhat maga után [31]. A biometrikus sablonok védelmével kapcsolatos lehetőségekről a későbbiekben még lesz szó a tanulmányban.

A biometrikus sablonok tárolásával kapcsolatos problémák, illetve a jogi szabályozásokból adódó nehézségek mellett más jellegű kihívások is felmerülnek. Ilyen például a kellő megbízhatóság elérése, ugyanis a biometrikus rendszerek pontossága sosem 100%-os. Az előforduló hibák két fő csoportra oszthatók, hibás elfogadásokra (FAR, false acceptance rate) és hibás visszautasításokra (FRR, false rejection rate). Hibás elfogadás alatt azt értjük, amikor a tárolt biometrikus sablonhoz tartozó személy nem egyezik meg azzal a személlyel, akinek a biometrikus sablonját ellenőrizzük, a rendszer mégis egyezést jelez, tehát jogosulatlan személynek ad hozzáférést. Ezzel szemben hibás visszautasításról akkor beszélünk, amikor a valóságban ugyanahhoz a személyhez tartozó biometrikus sablonokat vetünk össze, a rendszer mégis eltérést jelez, tehát jogosult személytől tagadja meg a hozzáférést. A hibás elfogadás úgy csökkenthető, ha a rendszer a biometrikus sablonok esetén elvárt egyezési küszöbértéket növeli (tehát csak nagyobb egyezés esetén ad hozzáférést), míg a hibás elutasítás épp ellenkezőleg, az elvárt egyezési küszöbérték csökkentésével méréskelhető. Ebből következik, hogy a két metrika csak egymás kárára javítható, ahogy azt a Ábra **6**Ábra **6**. ábra is mutatja. A megfelelő egyezési küszöbérték megválasztása azon múlik, hogy a felhasználók kényelmét (alacsonyabb

FRR és magasabb FAR) vagy a biztonságot akarjuk növeli (alacsonyabb FAR és magasabb FRR).



Ábra 6 - **A biometrikus azonosító rendszerek pontossága**

2.2 Az arcfelismerés alkalmazási területei

Mint ahogy fentebb említettük az arcfelismerést széleskörűen alkalmazzák mind az állami, mind a magánszektorban.

Az állami szektorban motiválja az arcfelismerés használatát nagy arcképes adatbázisok rendelkezésre állása, például személyi igazolvány képek, vezetői engedélyhez készült képek vagy bűnözői nyilvántartásban tárolt képek formájában (bár adatvédelmi okok miatt ezeknek az arcfelismerésre való alkalmazása nem minden esetben lehetséges). Például a korábban elítélt vagy körözött bűnözőkről készült arckép adatbázis használható határátkelésnél, ahol az utazók arcképeit vetik össze körözött bűnözők arcával. Másik lehetséges használat az ABC kapuk (Automated Border Control vagy Automatizált Határellenőrzési Rendszer) használata, mely során az utazók arcképeit az általuk felmutatott útlevélben található arcképpel vetik össze a forgalom gyorsítása céljából [7].

Kanadában kaszinókban is kísérleteztek arcfelismerés alkalmazásával, hogy távol tarthassák az ismert csalókat vagy kitiltott személyeket, vagy azokat, akik önszántukból iratkoztak fel egy tiltólistára, hogy így segítsék szenvedélybetegségük kezelését [7]. További lehetséges állami felhasználási terület a drónok arcfelismerő technológiával való felszerelése, mely lehetővé teheti számukra bizonyos kiszemelt emberek követését.

Hasonlóan katonai célú alkalmazással is kísérleteznek, például olyan speciális szemüvegek fejlesztésével, melyek képesek arcfelismerésre, s ez által nagyon gyorsan segíthetnek viselőjüknek kiszúrni akár nagy tömegben is az általuk keresett személyeket, például terroristákat [7].

Az arcfelismerés alkalmazása a privát szektorban is rendkívüli gyorsasággal terjed. 2008-ban a Lenovo dobott piacra olyan laptopot, mely jelszó helyett arcfelismerést is lehetővé tett a felhasználóknak, hogy így férjenek hozzá fiókjukhoz, de ma már mobiltelefonok képernyőzárának feloldására is bevett szokás arcfelismerést alkalmazni [7]. Szintén mindennapi jelenséggé váltak bizonyos online cégek által alkalmazott arcfelismerési szolgáltatások is, mint például a Facebook általi automatikus taggelés a felhasználók arcképei alapján.

Létezik olyan cég is, amelyik arcfelismeréssel rendelkező biztonsági kamerákat forgalmaz, melyek például képesek jelenteni, ha egy bizonyos személy, vagy adottnál több ember jelenik meg a helyszínen, sőt, akár belépés ellenőrzés megvalósítására is képesek lehetnek [7]. A kiskereskedelemben is megtalálható a technológia, itt például boltok használják arra, hogy vásárlóikról demográfiai adatokat vagy vásárlási szokásaikkal kapcsolatos adatokat gyűjtsenek. A banki szektor egyik érdekes alkalmazásával egy amerikai cég kísérletezett, mely ATM gépekben alkalmazott arcfelismerést a felhasználók hitelesítésének biztonságosabbá tételéért [7].

3 Arcfelismerési eljárások

A fejezetben bemutatjuk az elterjedt modern arcfelismerési rendszereket, valamint az ezek alapját képező architektúrákat, megoldásokat. Mivel a legkorszerűbb arcfelismerési rendszerek mély tanulási eljárásokra épülnek, ezért először a mély tanulással kapcsolatos fontosabb fogalmakat és eljárásokat mutatjuk be.

3.1 Arcfelismerés mély tanulással

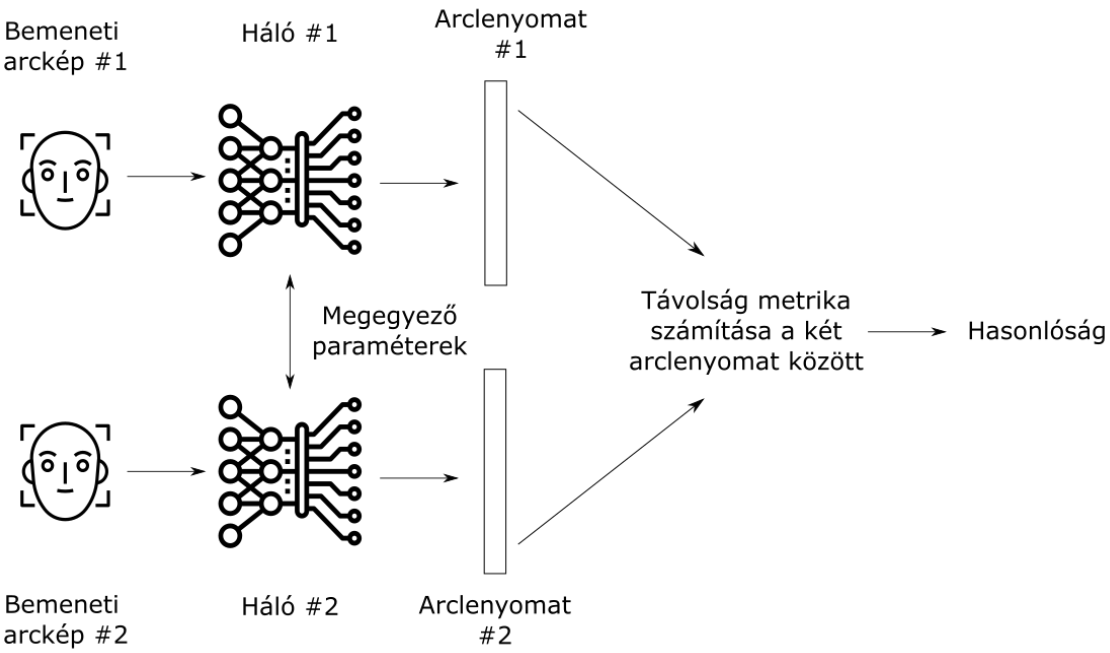
A gépi tanulásban sokszor van szükség adatpontok összehasonlítására valamilyen köztük értelmezett távolság alapján, hogy eldönthessük róluk, hogy azonos csoportba tartoznak-e valamilyen szempontból vagy sem (pl. azonos emberhez tartozik-e két kép vagy sem). Erre jó példa lehet az adatok csoportosításának problémája (clustering), sokszor azonban nehéz megfelelő metrikát találni az adatpontok jellemzésére, főleg, ha olyan komplex, nem-strukturált és erősen kontextusfüggő (pl. megvilágítás, arc szöge) adatokról van szó, mint az arcokat ábrázoló képek.

A deep metric learning (magyarul mély metrika tanulás) egy olyan felügyelt tanítási eljárás melynek célja, hogy a feladattól függően automatikusan előállítson olyan megfelelő metrikát az adatpontokhoz, mely metrika aztán felhasználható különböző feladatokra, például az adatpontok osztályozására (az előre megadott osztályok közül melyikbe tartozik egy adatpont) vagy klaszterezésére (az adathalmaz elemeinek melyik elkülöníthető csoportjába tartozik egy adatpont).

A fentiek értelmében a deep metric learning az arcfelismerés során is alapvető fontosságú. Ez esetben digitális arcképekről kell egy arcokat jellemző metrikát előállítani oly módon, hogy az ugyanahhoz az emberhez tartozó arcleíró vektorok (későbbiekben: arclenyomatok) távolsága kicsi legyen, míg a különböző emberek között nagy. Általában meghatározható egy küszöbérték is, mely alatti távolság esetén azt mondjuk, hogy két arclenyomat vektor ugyanahhoz az emberhez tartozik, míg a küszöbnél nagyobb távolság esetén azt mondjuk, hogy különböző emberhez tartozik a két lenyomat. Ez általában nem tökéletes, de beállítható a küszöb úgy, hogy a téves azonosítások aránya megfelelően kicsi legyen. Az alkalmazható távolság metrikákról részletesebben a későbbiekben lesz szó.

3.1.1 Arc azonosítás sziámi hálókkal

Az arclenyomat vektorokat létrehozó mély neurális hálók tanítása általában sziámi hálókkal történik [6]. Sziámi hálónak azt az architektúrát nevezzük, melyben kettő vagy több, egymással teljes mértékben megegyező neurális háló található, melyben, ha bármelyik háló bármelyik (súly) paramétere megváltozik, akkor ezt a változást az összes többi háló „tükrözi", azaz a hálók minden pillanatban megegyeznek. Ahogy a 4. ábrán látható, a sziámi hálók minden bemenetéhez tartozik egy háló, mely a bemenetből egy mély metrikát állít elő (pl. arclenyomat vektor), majd ezeket a mély metrikákat valamilyen távolság metrika szerint összevetve a háló a kimenetére kiszámítja, hogy mennyi a hasonlóság a két bemenet között. A sziámi háló architektúra lényege, hogy tanítás során a paramétereket úgy módosítjuk, hogy azonosnak tekinthető bemenetekre a háló nagy hasonlóságot, míg különböző osztályba tartozó bemenetek esetén a háló kis hasonlóságot jelezzen.



Ábra 7 – Sziámi hálók működésének bemutatása arcfelismerési példán keresztül.

Sziámi hálókat általában hitelesítési vagy azonosítási feladatok megoldására szoktak alkalmazni, mint például aláírások hitelességének vizsgálata, gyógyszerek összehasonlítása (pl. ugyanaz a felírt gyógyszer látható-e két képen) vagy az arcfelismerés. Az arcfelismerésre használt legnépszerűbb hálók, mint a FaceNet, a VGGFace vagy a ResNet is mind sziámi háló architektúrában lettek betanítva [6].

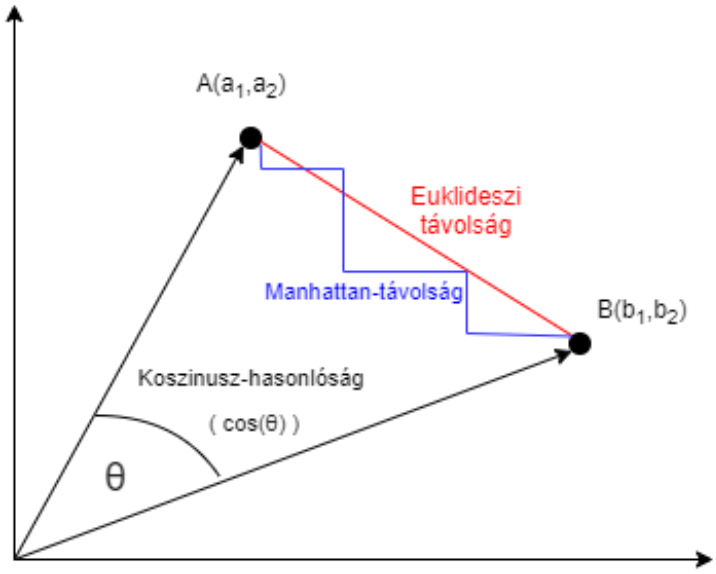
A sziámi hálók egyik legfőbb előnye az, hogy sokkal kevesebb tanító adatot igényelnek, mint más hálók, és a teljesítményük is jellemzően jobb. Ez a tulajdonság főleg olyan feladatok esetén előnyös, amelyben kevés tanító adat áll rendelkezésre osztályonként, vagy nagyon sok osztály van, esetleg nem lehet előre tudni minden lehetséges osztályt. Elképzelhető, hogy ezen helyzetek mind egyszerre fennállnak.

Az arcfelismerés klasszikus alkalmazásai tipikusan ilyen területnek számítanak, hiszen egy modellt nem lehet az összes létező emberi arcon (ahol egy ember arc modellje jelent egy osztályt) betanítani. A helyett, hogy betanítanánk a modellt arra, hogy ismerje fel a tanító adathalmazban lévő konkrét személyek arcait, és minden más arcot soroljon egy gyűjtő „egyéb" kategóriába, sokkal célszerűbb, ha arra tanítjuk meg a modellt, hogy el tudja dönteni két képről, hogy ugyanaz az ember látható-e rajta, függetlenül attól, hogy az adott ember arcáról szerepelt-e kép a tanító adathalmazban. Ez rugalmasabb adatbázis kezelést is lehetővé tesz az éles rendszerekben. Tehát sziámi hálókkal a cél olyan modellek tanítása, melyek két (vagy több) bemenet esetén el tudják dönteni, hogy a bemenetek azonos osztályhoz tartoznak-e vagy sem. Például egy arcfelismerő rendszer el tudja dönteni, hogy két arcképen ugyanaz az ember látható-e vagy sem, egy aláírás felismerő rendszer el tudja dönteni, hogy két aláírás megegyezik-e vagy az egyik hamisítvány és így tovább.

Az arcfelismeréshez használt arclenyomat vektorokat létrehozó sziámi hálók tanítására többek között egy úgynevezett „triplet loss" veszteségfüggvényt alkottak meg a kutatók. E veszteségfüggvény használatának lényege az, hogy tanítás során három sziámi háló bemenetére mindig három különböző arcképet adunk. E három arcképből az elsőt kinevezzük referenciának, a másodikat pozitív példának (mely ugyanannak az embernek az arcát ábrázolja, mint a referencia), a harmadikat pedig negatív példának (mely másik ember arcát ábrázolja). A tanítás során a neurális háló paramétereit úgy módosítjuk, hogy olyan arclenyomat vektorokat állítson elő a bemeneti képekből, hogy a referencia és a pozitív példáról készült arclenyomatok távolsága minél kisebb, míg a referencia és a negatív példa közötti távolság minél nagyobb legyen. Kellő mennyiségű tanító kép és megfelelően választott kép hármasok használata esetén elérhető, hogy a neurális háló megtanuljon általánosítani, vagyis olyan arcképekről is megfelelő arclenyomat vektorokat hozzon létre, amelyeket nem használtunk a tanítás során.

3.1.2 Távolság és hasonlóság metrikák arclenyomatokhoz

Az arclenyomatokat az origóból egy adott pontba mutató sokdimenziós vektorként értelmezve a vektorok végpontjai közötti távolság, valamint a vektorok közötti hasonlóság számos módon meghatározható. Jelen tanulmányban a három leggyakrabban használt metrikát mutatjuk be a teljesség igénye nélkül. A bemutatásra kerülő metrikákat az 5Ábra 8. ábra szemlélteti. Bár ennél lényegesen több metrika is alkalmas lehet arclenyomatok összehasonlítására (több tíz közül is választhatunk), most a leggyakrabban használtak közül mutatunk be néhányat az érthetőség kedvéért.



Ábra 8 - Az Euklideszi- és Manhattan-távolság, illetve a Koszinusz-hasonlóság szemléltetése.

3.1.2.1 Euklideszi távolság

Két pont távolsága alatt legtöbb esetben a két pont euklideszi távolságát értjük. Gépi tanulás esetén jól használható módszer, ugyanis gyors működést eredményez és releváns információt szolgáltat a rendelkezésre álló adatok viszonyáról. Ennek általános számítási módja n dimenzió esetén:

$$d_{euc}(A,B) = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 + .. + (a_n - b_n)^2},$$

ahol A = $(a_1, a_2, a_3, .. a_n)$ és B = $(b_1, b_2, b_3, .. b_n)$.

Az Ábra 8. ábra alapján felismerhető, hogy ez a metrika két pont közötti legkisebb távolságot adja meg. Az arcfelismerési eljárások esetén ezt a módszert használják a

legtöbb esetben. Ez a gyakorlatban úgy működik, hogy meghatározunk egy küszöbértéket, amely azt a távolságot jelenti, amelyen belüli ismert arclenyomatokat vizsgáljuk az új minta körül. Az arcfelismerés során megnézzük, hogy a vizsgált tartományon belül melyik arcról van a legtöbb minta, majd ebből következtetünk a vizsgált arclenyomatra.

3.1.2.2 Manhattan távolság:

A Manhattan-távolság, vagy más néven L1 távolság esetén a koordináta különbségek abszolútértékének az összegét vizsgáljuk. Általános számítási módja n-dimenzió esetén:

$$d_{man}(A,B) = \sum_{i=1}^{n} | a_i - b_i |,$$

ahol A = $(a_1, a_2, a_3, .. a_n)$ és B = $(b_1, b_2, b_3, .. b_n)$.

Megfigyelhető, hogy ezt a módszert használjuk a mindennapok során is, gondoljunk csak a városi közlekedésre, ahol a kiinduló pontból a végpontba a legtöbb esetben nem tudunk légvonalban eljutni, ami a legrövidebb távolságot jelentené.

Arcfelismerési problémák esetén kevésbé elterjedt, mint az euklideszi távolság mérése, de bizonyos esetekben érdemes kipróbálni mindkét metrikát (pl. homomorfikus titkosítás esetén hasznos lehet az egyszerűbb kiszámíthatóság).

3.1.2.3 Koszinusz hasonlóság

A koszinusz hasonlóság az eddigiektől eltérő megközelítés. Nem két pont közötti távolságot számol, hanem a két pontba mutató vektor hasonlóságát vizsgálja. Ez a hasonlóság a két vektor által bezárt szög koszinuszát jelenti. Értelemszerűen, ha a két vektor megegyezik, az általuk bezárt szög 0 fok, így a hasonlóság 1 értékű lesz. Amennyiben teljes mértékben különbözik a két vektor, a hasonlóság -1 értéket vesz fel.

Általános számítási mód n-dimenzió esetén:

$$\cos(\theta) = \frac{A * B}{||A||\,||B||} = \frac{\sum_{i=1}^{n} a_i b_i}{\sqrt{\sum_{i=1}^{n} a_i^2}\,\sqrt{\sum_{i=1}^{n} b_i^2}}$$

ahol A = $(a_1, a_2, a_3, .. a_n)$, B = $(b_1, b_2, b_3, .. b_n)$.

A módszer nem veszi figyelembe a vektorok hosszát, csak az általuk bezárt szöget. Így maximális hasonlóság áll fent egy V vektor és egy $c * V$ vektor között is, ahol c tetszőleges konstans.

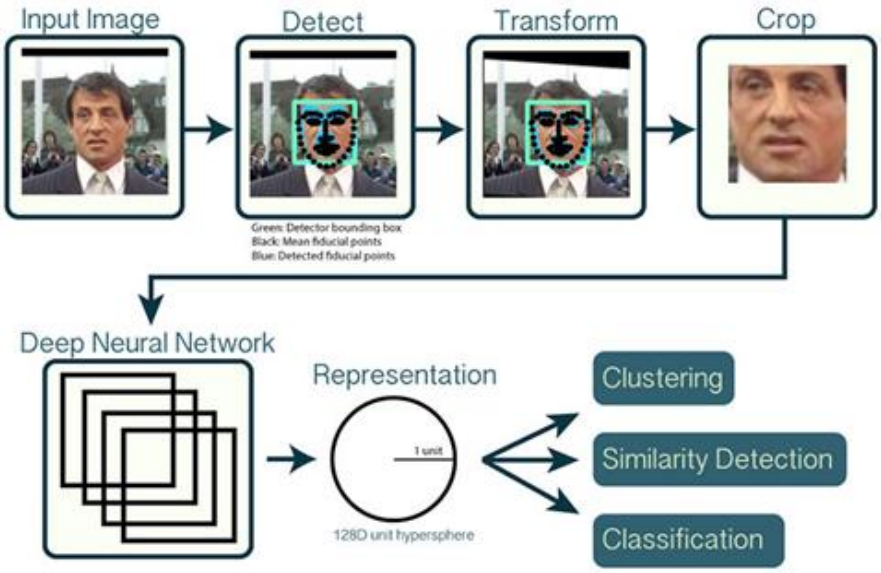
Egy hasonlósági küszöbérték definiálásával a módszer jól használható arc hitelesítésére, amennyiben rendelkezünk referencia képekkel a keresett arcról.

3.1.2.4 További metrikák

A bemutatott metrikákon túl számos hasonlósági és távolság mérésére alkalmas formula létezik. Ezek közül megemlítendő a Minkowski-távolság, amely az euklideszi és a Manhattan távolságok általánosított formája, valamint a Chebyshev-távolság, amely két vektor esetén az összetartozó koordináta párok közötti legnagyobb távolságot jelenti.

3.2 Arcfelismerő rendszerek bemutatása

A modern arcfelismerési eljárások néhány, jól elkülöníthető lépésből állnak, amelyeket a következő 6. ábra mutat be.



Ábra 9 – Arcfelismerés folyamata. A bemeneti képen meghatározzuk az arc pozícióját és orientációját, majd amennyiben szükséges, különböző transzformációkkal frontális képpé alakítjuk. A csak az arcot tartalmazó képből egy neurális háló létrehozza az arcot leíró struktúrát, amelyet különböző módszerekkel (pl. osztályozó algoritmusokkal) vizsgálhatunk arcfelismerés szempontjából. (ábra forrása: [13])

Röviden összefoglalva, a bemeneti képen egy arc detektorral megkeressük az arcot, majd különböző transzformációkkal igazítjuk. Ennek célja, hogy az arc leginkább úgy nézzen ki, mintha szemből fényképezték volna le. Ezt követően csak az arcot tartjuk meg (némi kerettel), amelyet átadunk egy neurális hálónak. Ez a háló egy arclenyomat vektort hoz létre, amely az arc fő jellemzőit írja le. Ezt követően különböző osztályozó algoritmusokat felhasználva az arclenyomat vektorok alapján meghatározható a keresett személy.

3.2.1 Arcfelismerő programozási könyvtárak

Ezen logika mentén működnek a Google és a Facebook megoldásai is. A Facebook kutatói 2014-ben publikálták a *DeepFace [18]* nevű arcfelismerési rendszerüket. Az arc leírása egy 9 rétegű neurális hálóval történik, amely több mint 120 millió paramétert tartalmaz. A háló tanításához 4 millió arcot ábrázoló képet használtak, mintegy 4 ezer személyről. A módszer 97 % körüli pontossággal azonosította az egyes arcokat a *Labeled Faces in the Wild (LFW) [23]* teszt adathalmazon, mely összemérhető az emberi képességekkel. Az LFW a leggyakrabban használt „benchmark" adathalmaz arcfelismerési technológiák teljesítményének összehasonlítására, mely több, mint 13000 névvel felcímkézett arcképet tartalmaz (az adathalmazban 1680 olyan ember van, akihez több különböző kép is tartozik).

A Google kutatói egy évvel később, 2015-ben mutatták be a *FaceNet [19]* nevű arcfelismerési rendszert. Az arclenyomatot teljes egészében egy neurális háló hozza létre, nincs szükség PCA-ra (Principal Component Analysis, magyarul főkomponens analízis) a kimeneten, valamint SVM-re a klasszifikációhoz, mint ahogy az a korábbi hálók esetében volt. Több architektúrát mutattak be, ezek közül a különböző méretű *Inception* modellek és a *Zeiler&Fergus* architektúra bizonyult a legpontosabbnak. A *Zeiler&Fergus* alapú háló nagyságrendileg 140 millió paramétert tartalmaz, ezzel szemben az *Inception* „csak" 7.5 milliót, így egy megfelelően választott modell mérettel az utóbbi alkalmas lehet okostelefonnal történő arcfelismeréshez a gyorsasága miatt. A tanító adathalmaz 260 millió képet tartalmazott, mintegy 8 millió különböző személyről. A háló tanítása nagyságrendileg 1000-2000 órát vett igénybe CPU kluszteren. A rendszer a legnagyobb méretű *Inception* modellel a már említett LFW adathalmazon 99.6 %-os pontosságot ért el.

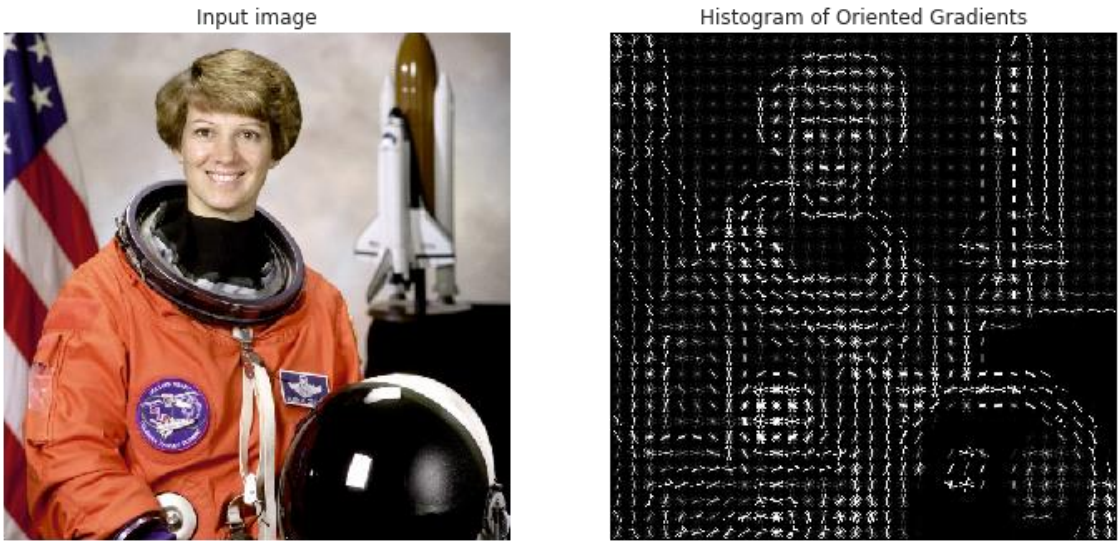
Az előbbiekben említett két megoldással szinte lehetetlen versenyezni, gondoljunk csak a szükséges tanító adathalmazra, a nagy hardverigényre vagy az egyedi arc detektorokra, transzformációkra. Azonban létezik számos nyílt forráskódú könyvtár, amelyek arcfelismerő rendszert kínálnak, az előbbiekkel összemérhető teljesítménnyel. A *face_recognition [17]* könyvtár az arc megtalálásához és leírásához is a *dlib* [14] könyvtár algoritmusait használja. Az arcot leíró neurális háló a Microsoft által publikált ResNet-34 [21] architektúráján alapszik. Az *InsightFace* [16] Python könyvtár az ArcFace [20] architektúra alapján írja le az arcjellemzőket, az arc megtalálásához pedig a RetinaFace [25] arc detektort használja. Emellett az OpenCV [15] is kínál különböző arc detektorokat, valamint keretrendszert egyedi arcleíró modell használatára. A következőkben bemutatjuk ezen nyílt forráskódú megoldások alapjait.

A továbbiakban azokat a megoldásokat ismertetjük, amelyek a legelterjedtebb, nyílt forráskódú rendszerek alapját képezik. Léteznek további algoritmusok melyek zártak, gondolunk itt például a VGGFace és a DeepID arcleíró modellekre, azonban tanulmányunkban a mindenki számára elérhető, ingyenes megoldásokra helyezzük a hangsúlyt.

3.2.2 Arcdetektálás

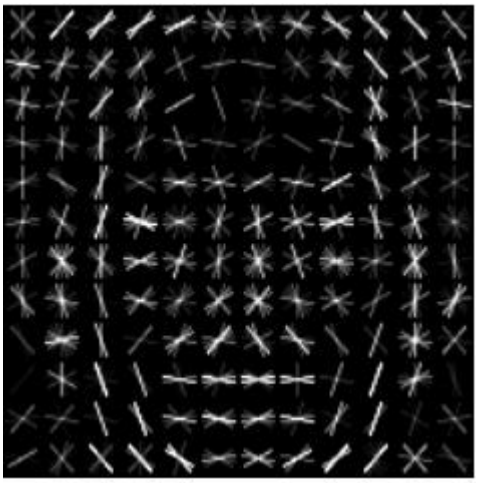
Első lépésként meg kell találni az arcot a bemeneti képen. Erre a feladatra leggyakrabban a dlib és az OpenCV arcdetektálási megoldásait alkalmazzák. A következőkben bemutatjuk az egyes módszereket, illetve ezek előnyeit, hátrányait.

A *dlib* könyvtár két technikát kínál az arcfelismerésre. Az első, és egyben egyszerűbb módszer, az úgynevezett *Histogram of Oriented Gradients* (HOG). Alapvetően bármilyen típusú, formájú objektum detektálására alkalmas lehet, gyorsasága miatt alkalmazzák arcfelismerési eljárásokhoz. Az algoritmus a bemeneti kép összes pixelére számol egy gradiens értéket, amely az adott pont élszerűségének a mértékeként értelmezhető. A kapott élek alapján a hasonló alakzatok felismerhetők, ahogyan az a következő Ábra 10. ábrán is látható.



Ábra 10 - HOG algoritmus által megtalált élek (ábra forrása: [26])

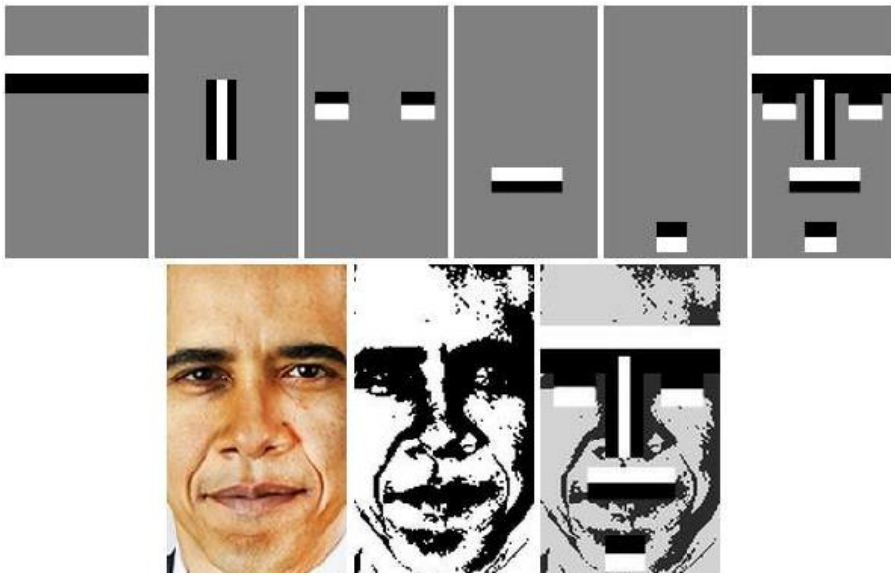
A dlib új verziójában már elérhető egy neurális háló alapú arc detektor is, amelya *max-margin object-detection (MMOD)* [36] algoritmus alapján betanított HOG szűrőt használja az arc detektálására. A szűrő tanításához az egyes képeket fel kell címkézni, hogy hol található rajta az arc. Az algoritmus a tanulásnál egy csúszóablakot használ az arc megtalálásához a címkék alapján. Ennek előnye, hogy így a tanulás során a kép minden részletét felhasználja. A szűrő tanítás után általánosítva tartalmazza a keresett objektumhoz tartozó lehetséges gradiens értékeket, éleket. Egy ilyen lehetséges arc leírást mutat be a következő Ábra 11. ábra.



Ábra 11 - Az emberi arc általános leírása a HOG szűrő alapján a tanítás után (ábra forrása: [36])

Miután a bemeneti képre kiszámoltuk az egyes gradiens értékeket, a szűrővel összevetve eldönthető, hogy a keresett objektum – jelen esetben arc - szerepel-e a képen vagy sem.

Az OpenCV is egy klasszikus és egy neurális háló alapú algoritmust kínál az arc megtalálásához. A Haar-Cascade - a HOG-hoz hasonlóan – egy általános objektumfelismerő eljárás. Az algoritmus úgynevezett Haar típusú képjellemzőket használ. A képjellemzők egy bináris ablak segítségével vizsgálják az egyes képrészleteket. Az ablak fehér része alá eső pixelek intenzitásából kivonják a fekete rész alá eső intenzitásokat. Hasonlóságról akkor beszélhetünk, ha az eredmény egy nagy értékű pozitív szám, amennyiben pedig negatív, úgy különbözőség áll fent. Ezzel a technikával ugyanúgy él jellegű képrészeket lehet detektálni, mint a HOG eljárással.



Ábra 12 - Haar jellemzők arcdetektáláshoz (ábra forrása: [27])

Mivel számos képjellemzőt kell vizsgálni, a hasznos jellemzők kiválasztása egy tanító adatbázis alapján történik. Az algoritmus gyorsítása érdekében további szűrőket alkalmaznak, amelyek az arcot nem tartalmazó részletek egy részét ki tudják szűrni a vizsgált tartományból.

Az OpenCV egy konvolúciós háló alapú modellt is támogat az arc megtalálására, amely az SSD (Single Shot Detector) architektúrán alapúl. A módszer egyenlő méretű tartományokra osztja fel a képet, amelyek egyenként felelősek a keresett objektum detektálásáért.

Arc detektor / **Tulajdonságok**	**HOG**	**Dlib-NN**	**Haar-Cascade**	**OpenCV-NN**

Valós idejű működés CPU-n	**X**	**-**	**X**	**X**
Valós idejű működés GPU-n	**X**	**X**	**X**	**X**
Arc orientációjától független működés	**-**	**X**	**-**	**X**
Zajra robosztus	**-**	**X**	**-**	**X**
Bemeneti kép méretétől független	**-**	**-**	**X**	**X**

Táblázat 1 - Az egyes arc detektorok összehasonlítása.

3.2.3 Arcjellemzők kinyerése

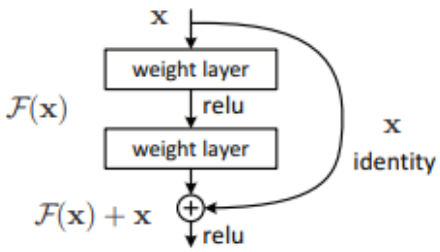
Ezek a vektorok a használt architektúrától függően jellemzően lehetnek 128- vagy 512-dimenziósak, de akár ettől eltérő méretűek is. Az arcfelismerés esetén a metrikus tulajdonságukat használjuk fel, vagyis, hogy az ugyanazt az arcot leíró vektorok az adott dimenziós térben egymáshoz közel esnek. A távolság mérésére alkalmas módszerek már bemutatásra kerültek a 3.1-es fejezetben.

Az OpenFace [13] egy deep learning alapú arcleíró modell a Google FaceNet architektúrái alapján. Az OpenFace az egyik leggyakrabban alkalmazott modell az utóbbi időben, mivel hasonló pontosságot biztosít, mint a nem nyílt forráskódú algoritmusok.

A modell több mint 500 000 képen volt tanítva, amelyek 11 ezer különböző személyről készültek, ehhez az alkotó két adathalmazt használt fel (*CASIA-WebFace [32], FaceScrub [24]*). A modell kimenetként 128-dimenziós arclenyomat vektort hoz létre. Fontos megemlíteni, hogy magasabb dimenziójú vektorok esetén a háló tanítási ideje lényegesen megnőne, lassabb, de nem feltétlenül pontosabb működést eredményezve, amely valós-idejű alkalmazások esetén hátrányt jelentene.

Egy további, bárki számára elérhető architektúra arclenyomat vektorok generálására a már említett *dlib* könyvtár ResNet-34 alapú architektúrája. Ez a módszer a Microsoft ResNet megoldásán [21] alapul, kisebb változtatásokkal a gyorsabb működés érdekében. Az architektúra úgynevezett reziduális blokkokból épül fel, amely a 10. ábrán látható.

Ezen blokkok használatával kiküszöbölhetők a nagyon mély hálók esetén egyes rétegek által okozott hibák, az információk előre csatolása által.



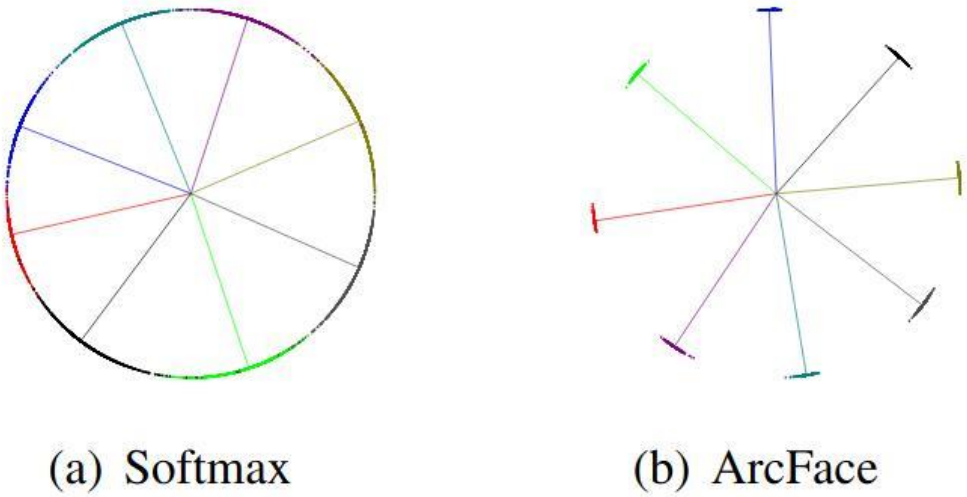
Ábra 13 – A ResNet hálók által alkalmazott reziduális blokkok sematikus vázlata. Az előrecsatolás miatt egy identitásfüggvény is megvalósítható egy ilyen blokkal, amely eredményeként a blokk nem befolyásolja negatívan a háló működését, legrosszabb esetben olyan, mintha az átugrott réteg „ott sem lenne" (ábra forrása: [21]).

Azt gondolnánk, hogy a neurális hálók mélységének a növelése pontosabb működést eredményez, azonban ez a gyakorlatban nem mindig igaz, sőt, valójában bizonyos esetekben a túl komplex hálók nagyobb hibát eredményeznek. A reziduális blokkok erre kínálnak megoldást, ugyanis ezek használatával növelhetjük a háló mélységét anélkül, hogy az előbb említett hibás működés bekövetkezne. A tanítás során, ha a háló nem tudja az adott rétegeket felhasználni, az előrecsatolás miatt egy ilyen blokk identitásfüggvényt valósíthat meg. Ez a gyakorlatban azt jelenti, hogy amit az adott blokk a bemenetén megkap, azt továbbítja a kimeneten. Ennek következtében a nem használt rétegek nem rontják a háló pontosságát.

A dlib modell tanításához nagyságrendileg 3 millió arcot használtak, többek között a *FaceScrub* [24] és a *VGG face* [33] adathalmazok felhasználásával. A modell 128-dimenziós arclenyomat vektorokat hoz létre az arc egyedi jellemzői alapján.

Az ArcFace [20] architektúra az eddigiektől eltérő megközelítést használ. Egy új veszteség-függvényt, az *Additive Angular Margin Loss*-t használták a háló tanításához, az eddigi *softmax* helyett. Ez a módszer a gépi tanuláshoz előállított lenyomatokra van optimalizálva, így arc leíráshoz is jól használható. A modell így robosztusabb például az arc pozíciójára vagy a kor változására. Az architektúra célja, hogy az egy személyhez tartozó arclenyomatokat a lehető legközelebb hozza egymáshoz, míg az eltérő személyekhez tartozókat pedig távolra. A következő, 11. ábrán látható, hogy a softmax veszteségfüggvényhez képest hogyan helyezi el a lenyomatokat az adott dimenziójú hipergömbön. Megfigyelhető, hogy a különböző arcokhoz tartozó lenyomatok egymástól

lényegesen nagyobb távolságra helyezkednek el. A modell 512-dimenziós lenyomatokat generál az arcról.



Ábra 14 – A Softmax és az ArcFace eljárással tanított hálók által generált arclenyomat vektorok elhelyezkedése egy hipergömbön (látható, hogy az ArcFace eljárással generált különböző emberekhez tartozó arclenyomat vektorok sokkal jobban elkülönülnek egymástól) [20]

Az ArcFace-t implementáló, nyílt forráskódú Python könyvtár az *InsightFace* [16] .A modellt több adathalmaz kombinálásával tanították be, ezek a *MS1M [35]*, *VGG2 [34]*, *CASIA-Webface [32]* és egyéb nem részletezett képek.

3.3 Embeddingek felhasználása azonosításra

Ezen módszerek segítségével tehát előállíthatunk arclenyomat vektorokat, amelyek alapján az arc azonosítható (pl. egy adatbázisból kikereshető, hogy ki van a képen). A következőkben bemutatjuk az arclenyomat vektorok használati módját.

Azonosításra elsősorban klaszterezés alkalmazható. Klaszterezésnek azt nevezzük, hogy az egyes emberek arclenyomat vektorai emberenként különböző csoportokba, más néven klaszterekbe rendeződnek az N dimenziós térben (pl. 128 dimenziós arclenyomat vektorok esetén 128 dimenziós térben). Például, ha öt különböző emberről tárolunk emberenként 10 arclenyomat vektort, akkor ideális esetben öt jól elkülönülő klasztert kapunk, klaszterenként 10 arclenyomat vektorral. Így amennyiben egy ember azonosítása szükséges, akkor a róla készült arclenyomat vektor távolságát kell megvizsgálni az egyes klaszterektől (pl. klaszter középpontoktól), s amelyik klaszterhez

a legközelebb van, vélhetően az ahhoz a klaszterhez tartozó személyről van szó. A távolság meghatározására általában valamilyen küszöb séma alkalmazandó. A küszöb séma alkalmazásakor az azonosítandó emberről készült arclenyomat vektor és az adatbázisban tárolt arclenyomat vektorok távolságát számítjuk ki először. Amennyiben a legkisebb távolság egy előre meghatározott küszöbérték alatt van, úgy a személyt azonosítottnak tekintjük (pl. ugyanahhoz a klaszerhez tartozik, amelyikbe a hozzá legközelebb eső embedding).

Az arcképek alapján készült arclenyomat vektorokat többféleképpen lehet hitelesítésre felhasználni. Az egyik legkézenfekvőbb megoldás az arclenyomatok osztályozása, ami azt jelenti, hogy megfelelő gépi tanulási módszerrel betanítunk egy modellt arra, hogy minden arclenyomatról meg tudja határozni, hogy az melyik emberhez (osztályhoz) tartozik.

Ezt a fajta tanítást felügyelt tanításnak nevezzük, mivel minden tanulópéldához, tehát minden arclenyomathoz a tanító adathalmazban, rendelkezésre kell álljon közvetlenül az osztálycímke, tehát az, hogy az adott arclenyomat melyik emberhez tartozik. Ezzel a tanítással lehetőségünk van arra, hogy a modell megfelelően általánosítson, azaz olyan arclenyomatról is helyesen meg tudja mondani, hogy melyik emberhez tartozik, amit nem használtunk fel a tanítás során. Az osztályozás kétféleképpen is történhet.

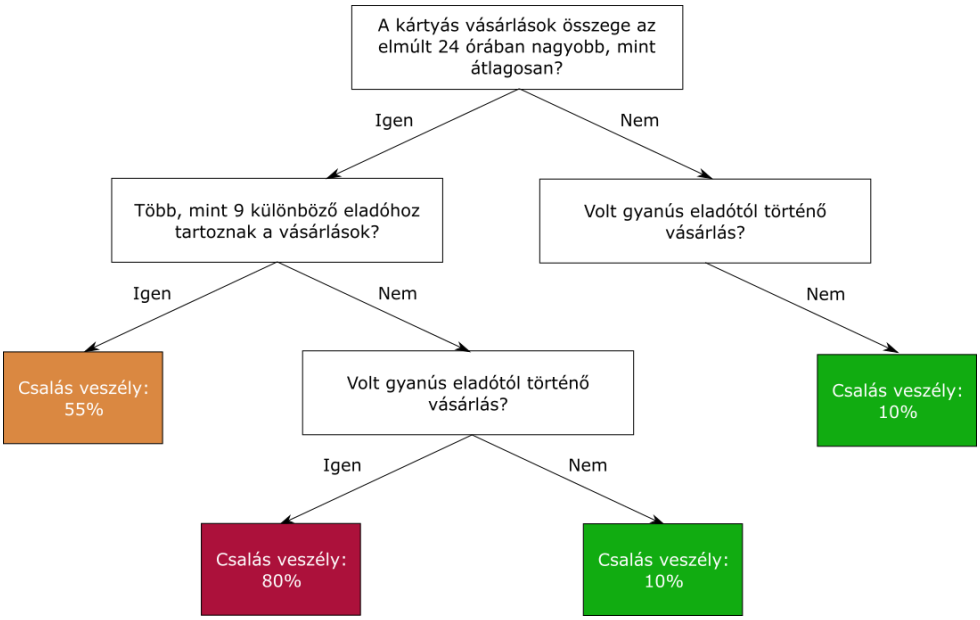
Egyrészt történhet úgy, hogy a gépi tanulási modell azt mondja, hogy egy adott arclenyomat egy konkrét emberhez tartozik-e vagy sem. Ez esetben bináris osztályozásról beszélünk, mivel két lehetséges kimenete van (ún. singleclass classification): az adott arclenyomat vagy az adott emberhez tartozik, vagy sem. Ilyen jellegű osztályozás esetén természetesen annyi bináris modellt kell betanítanunk, ahány különböző embert szeretnénk felismerni, ugyanis minden ember esetén saját bináris osztályozó modellre van szükség, s egy adott arclenyomathoz tartozó ember azonosítása úgy történik, hogy minden bináris modell ad egy predikciót az arclenyomat alapján, s az arclenyomat ahhoz az emberhez tartozik, akinek a prediktora a legnagyobb valószínűséget adja.

A másik lehetséges megoldás az, ha csak egy modellt tanítunk be, de az kettőnél több osztály megkülönböztetésére is képes. Ez esetben több osztályú osztályozásról beszélünk (ún. multiclass classification). Az ilyen modell tanítása szintén felügyelt tanítást igényel, tehát a tanítás során rendelkezésre áll minden arclenyomathoz a hozzá

tartozó ember címkéje, s a tanítás végére a modell megtanul általánosítani, azaz képes lesz egy olyan arclenyomatról is megmondani, hogy melyik emberhez tartozik, amely arclenyomat nem szerepel a tanító adathalmazban.

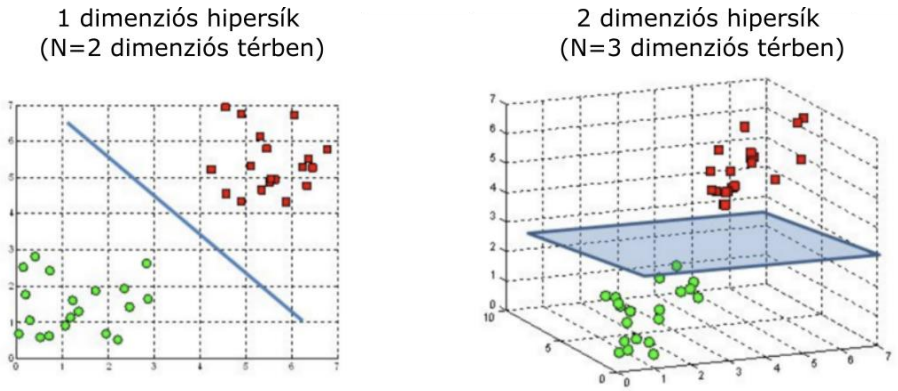
Mindkét eljárás csak olyan emberek felismerésére lesz képes, akikről volt tanulópélda a tanító adathalmazban, tehát új "idegen" emberek arclenyomatát nem lesznek képesek felismerni. Több gépi tanulási technika is létezik osztályozásra, ezek közül az alábbiakban bemutatunk hármat a teljesség igénye nélkül.

Az egyik legalapvetőbb osztályozási algoritmus a döntési fa, angolul decision tree. A döntési fák a tanító adatokat kettő vagy több halmazra bontják szét, az adatok különböző tulajdonságaira teljesülő vagy nem teljesülő feltételek alapján. A feltételek alapján szétválogatott adatok így a köztük lévő különbségek alapján külön csoportokra, azaz külön osztályokra válnak szét. A tanító adathalmazban nem szereplő adatpont osztályozása úgy történik, hogy a döntési fában szereplő feltételeket megvizsgálva az új adatpontra, eldönthető, hogy az melyik osztályba sorolandó. A döntési fák általánosítási képessége javítható, amennyiben ugyanabból a tanító adathalmazból kiindulva több véletlenszerűen betanított döntési fát hozunk létre. Ekkor az osztályozandó adatpontot minden döntési fa osztályba sorol, majd összesítjük a döntési fák szavazatait, s azt az osztályt tekintjük a végleges osztálynak, melyre a legtöbb szavazat érkezett. Ezt az algoritmust nevezzük véletlenszerű erdőnek (angolul random forest). Egy sematikus döntési fa látható a 12. ábrán.



Ábra 15 – Sematikus döntési fa bankkártyás csalások felderítésére.

Egy másik népszerű osztályozási algoritmus a tartó vektorgép, angolul support vector machine (SVM). Ennek célja, hogy olyan hipersíkot találjon az N dimenziós térben (ahol N az adatok tulajdonságainak számát jelöli), amely a legjobban elkülöníti a különböző osztályba tartozó adatpontokat. Például N=2 esetén egy 1 dimenziós egyenes, N=3 esetén egy 2 dimenziós sík stb (lásd 13. ábra). Előfordulhat sok esetben, hogy az adatok az eredeti tulajdonságaik alapján nem lineárisan elválaszthatók. Ekkor használható az úgynevezett kernel trükk, mely során magasabb dimenziójú térbe transzformáljuk az adatokat, amely térben már alkalmazhatunk lineáris szeparációt. Mivel az SVM egy hipersíkkal lényegében két részre osztja az adatokat, ezért egy SVM csak bináris osztályozásra használható. Több osztályos osztályozás esetén a “one versus one”, azaz “egy az egy ellen” eljárást kell alkalmaznunk, melynek során minden lehetséges osztály párra betanítunk egy SVM-et, majd mindegyik SVM predikcióját figyelembe véve egy adott adatpont abba az osztályba kerül, melyre a legtöbb predikció érkezik. Ebből következik, hogy n-osztályos osztályozási feladat megoldásához $\frac{n*(n-1)}{2}$ darab SVM betanítása szükséges, ami elég erőforrás igényes megoldás.



Ábra 16 – Support Vector Machine (SVM) alkalmazása adatpontok szétválasztására, osztályozására (az ábra forrása: [4])

Osztályozási feladatok megoldására lehetőségünk van neurális hálók alkalmazására is. Arclenyomatok osztályozására például úgy alkalmazható egy neurális háló, hogy különböző emberek arclenyomat vektorait beadva a hálónak, különböző neuronok aktivációs értéke lesz a legnagyobb a háló utolsó rétegében. Természetesen emiatt a legutolsó rétegnek mindig annyi neuronból kell állnia, ahány különböző embert akarunk osztályozni, tehát új ember hozzáadása esetén az egész hálót újra kell tanítani, viszont így több osztályos osztályozásra is kiválóan alkalmazhatók.

A neurális hálókkal kapcsolatban megjegyzendő, hogy sziámi architektúrájú neurális hálók is alkalmazhatóak hitelesítésre. Erre egy lehetséges példa egy olyan sziámi háló, mely egyszerre két arcképet fogad a bemenetén, s a kimenetén azt jelzi, hogy a két arckép azonos emberhez tartozik-e vagy sem. Hitelesítésre ezt olyan esetben lehetne alkalmazni, amikor a felhasználókról arcképet tárolunk, s akkor adunk hozzáférést egy felhasználónak a fiókjához, ha a róla készült arckép és az adatbázisban tárolt arckép a sziámi háló szerint ugyanahhoz az emberhez tartozik.

4 Az arcfelismerés nehézségei és kockázatai

Bár az arcfelismerés rendkívüli ütemben terjed, de alkalmazása nem jár kockázatok nélkül. Az alábbiakban a jelentősebb kockázatok és veszélyek kerülnek bemutatásra.

4.1 Részrehajlás, tömeges megfigyelés, adatvédelmi kockázatok

A deep learning alapú arcfelismerési eljárások a legtöbb esetben az emberével összemérhető eredményeket produkálnak. Azonban bizonyos esetekben arcfelismerő rendszerek alapján történő azonosítás téves eredményeket szolgáltathat. Egy friss amerikai kutatás [22], amelyben 189 arcfelismerő algoritmust vizsgáltak, rávilágított, hogy az afro-amerikai és ázsiai rasszba tartozó emberek esetén lényegesen nagyobb a fals pozitív találatok aránya. Ez a gyakorlatban azt jelenti, hogy bizonyos esetekben tévesen azt gondoljuk, hogy a keresett személy van a képen, pedig valójában nem. Ez jelentős probléma, gondoljunk csak arra, ha valakit egy videó miatt elítélnek, mert az arcfelismerő rendszer alapján szerepel a videón, pedig a valóságban nem volt köze a történtekhez. Valószínűsíthető, hogy a közeljövőben a probléma megoldása több kutatás központi témája lesz.

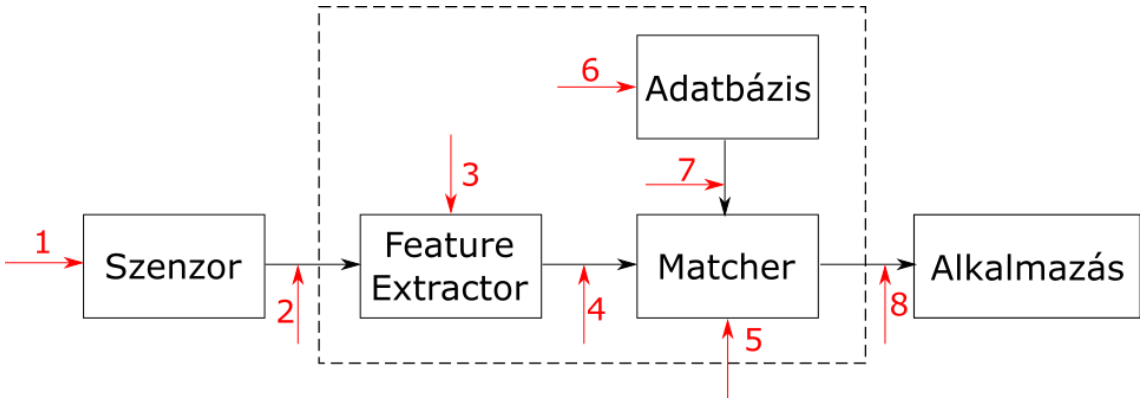
Szintén az arcfelismeréssel kapcsolatos probléma, hogy mivel alkalmazása se közvetlen fizikai kontaktust, se az emberek beleegyezését vagy együttműködését nem igényli, ezért tömeges megfigyelést tesz lehetővé. Ez a GDPR értelmében komolyan fenyegeti az emberek alapvető jogait és szabadságát, különös tekintettel a személyes adataik védelméhez való jogukra [45]. Továbbá például tüntetések bekamerázása, emberek követése stb. által a társadalom anonimitáshoz való jogát is veszélyezteti a technológia [46], hisz ilyen mértékű és pontosságú tömeges megfigyelést semmilyen korábbi biometrikus azonosítási eljárás nem tett lehetővé.

A helyzetet bonyolítja, hogy az arc egy egyedi, nagyon körülményesen megváltoztatható azonosító, amely alapján történő hitelesítés jelentős biztonsági kockázatokat rejt, és éppen ezért alkalmatlan a hitelesítésre. A legtöbb esetben hitelesítéshez valamilyen PIN kódot, jelszót vagy azonosító kártyát használunk. Ha

ezeket elfelejtjük, elveszítjük vagy illetéktelenek kezébe kerülnek, megváltoztathatjuk, vagy letilthatjuk azokat. Azonban, ha az arcot használjuk hitelesítésre, azt nem tudjuk megváltoztatni, ennek következtében nem tudjuk a hitelesítéshez szükséges információkat sem megváltoztatni. Ideális esetben a felhasználónak – csak úgy, mint egy jelszó esetén – lehetősége van különböző alkalmazásokhoz különböző arc leírásokat használni, valamint ezeket a leírásokat megváltoztatni, cserélni. Erre a problémára nyújt megoldást a későbbiekben bemutatásra kerülő biometrikus sablon védelmi eljárások alkalmazása.

4.2 Arcfelismerő rendszerek sebezhetőségi pontjai

További kihívást jelent az arcfelismerő rendszereket megfelelő védelemmel ellátni az esetleges támadások ellen. Általánosságban egy biometrikus azonosító rendszert 8 különböző ponton lehet megtámadni [40], ahogy azt a Ábra 17. ábra mutatja. A támadásokat alapvetően két csoportra oszthatjuk: külső és belső támadásokra. Külső támadás minden olyan támadás, amelyhez nincs szükség a biometrikus rendszer belső működésének befolyásolására, míg belső támadások esetén a támadó hozzáfér valamilyen rendszerelemhez (pl. adatbázis vagy kommunikációs csatornák).



Ábra 17 - Biometrikus azonosító rendszer lehetséges sebezhetőségi pontjai (1 – szenzor, 2 – bemeneti csatorna, 3 – feature extractor, 4 – feature extractor kimeneti csatorna, 5 – matcher, 6 – adatbázis, 7 – adatbázis kimeneti csatorna, 8 – kimeneti csatorna) (saját ábra [40] alapján)

1. **Szenzor bemenet hamisítása**. Egyes külső támadások a biometrikus rendszer szenzorának átverését kísérlik meg, mely során valamilyen hamis biometrikus mintát prezentál a támadó a szenzornak. Ilyen lehet egy hamis ujj, aláírás másolat, arc maszk, videófelvétel stb. Ezeket a támadásokat prezentációs támadásnak is nevezzük (presentation attack). Az ilyen támadások ellen

úgynevezett PAD (Presentation Attack Defense) rendszerekkel lehet védekezni, amik általában challenge-response elvre épülnek, vagyis valamilyen feladványt adnak a felhasználónak, és csak akkor adnak hozzáférést, ha a felhasználó sikeresen megoldja a kapott feladatot (pl. arcfelismerő rendszer esetén az illetőnek mosolyognia kell a belépéshez). Egy másik védekezési mód az úgynevezett "liveness detection" (15. ábra), melynek célja megállapítani, hogy elég életszerű-e a szenzorba érkező jel, vagy sem (pl. arcfelismerés esetén mélység érzékelésre is képes kamera alkalmazása).

2. A **szenzor megkerülésével** lehetséges úgynevezett visszajátszásos támadások végrehajtása is (replay attack). Visszajátszásos támadás során egy korábbi elfogadott biometrikus sablon másolatát juttatja a rendszer bemenetére a támadó, a szenzort megkerülve, így nyerve hozzáférést a rendszerhez. Egy lehetséges védekezési módszer a visszajátszásos támadások ellen, ha valamilyen időben változó információt vagy random számot is kell tartalmaznia minden beérkező üzenetnek, mellyel elkerülhető ugyanannak a beérkező csomagnak a többszöri feldolgozása.

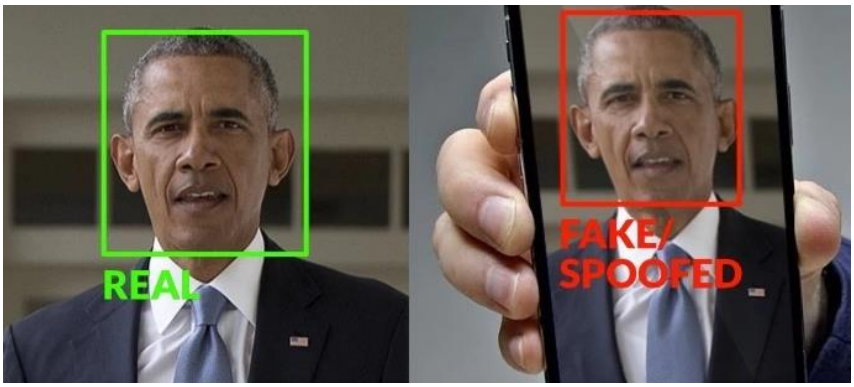
3. A **feature extractor** az az egység, melynek feladata a biometrikus sablon kivonása a szenzor bemenetből (pl. arclenyomat vektor előállítása az arcképből). Amennyiben a támadó képes befolyásolni ennek a rendszerelemnek a működését, úgy bármilyen bemeneti érték esetén bármilyen biometrikus sablont előállíthat, így képes jogtalan hozzáférést adni. Ezzel a támadással szemben egy kellően védett helyen tárolt, tamper-proof feature extractor használata nyújthat védelmet.

4. Amennyiben **a feature extractor kimeneti csatornáján** továbbított adatokat képes a támadó megváltoztatni, úgy lényegében képes a 3-as pontban említett támadással egyenértékű támadást végrehajtani, a továbbított biometrikus sablon megváltoztatása által. Ezzel a támadással szemben megfelelő kriptográfiai eljárásokkal (pl. titkosított kommunikáció) lehet védekezni.

5. A **matcher** feladata a bejövő biometrikus sablon összevetése a tárolt biometrikus sablonnal, hogy meghatározzon egy hasonlósági metrikát. Amennyiben a támadó hozzáfér a matcher-hez, úgy elérheti, hogy a matcher

által szolgáltatott hasonlósági metrika kellően magas vagy alacsony legyen (a támadó akaratától függően). Hasonlóan a 3-as támadáshoz, ebben az esetben is egy kellően védett helyen tárolt, tamper-proof matcher használata nyújthat védelmet.

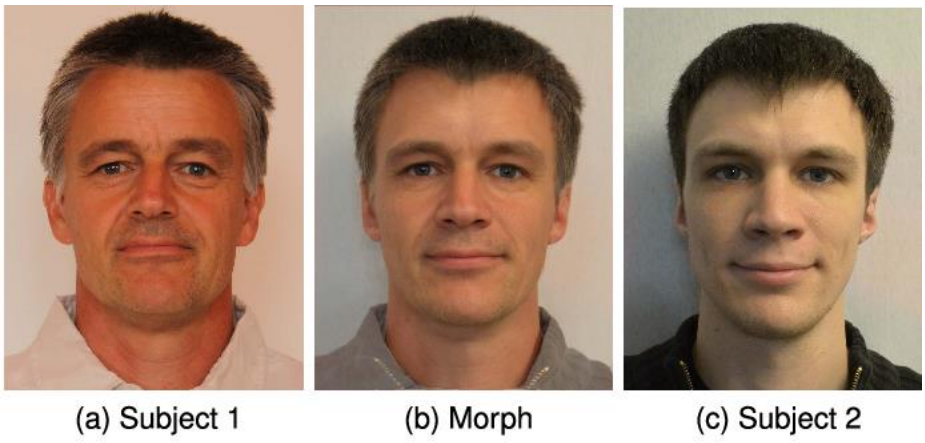
6. A **biometrikus sablonokat tároló adatbázis** szintén támadási pontot jelent. Amennyiben a támadó képes módosítani a tárolt sablonokat, úgy képes akár jogtalan hozzáférést adni egy külső személynek, vagy akár jogtalanul elutasítani a módosított sablonhoz tartozó személyt. Ez ellen a támadás ellen védelmet jelenthet az adatbázis biztonságos helyen tárolása, illetve egy „elosztott" sémájú adatbázis használata, melyben a felhasználók maguk tárolják a hozzájuk tartozó biometrikus sablont, melyet hitelesítéskor bemutatnak (pl. smart card-ok használatával).
7. **Matcher bemenetének manipulációja**. Amennyiben a támadó hozzáfér az adatbázis és a matcher közötti kommunikációs csatornához, úgy képes lehet a matcher-be érkező adatok megváltoztatására. Így akár az adatbázisból helyesen kiolvasott sablonokat is képes lehet akarata szerint megváltoztatni a matcher-be érkezés előtt. Ez ellen a támadás ellen a 4-es esethez hasonlóan szintén megfelelően titkosított kommunikációs csatorna használata nyújthat védelmet.
8. Végezetül amennyiben a támadó képes felülírni **a kimeneti csatornát**, úgy még egy tökéletesen működő biometrikus azonosító rendszer is teljesen kompromittálódhat. Ez ellen a támadás ellen is megfelelően titkosított kommunikációs csatornák alkalmazása nyújthat védelmet.



Ábra 18 – A „liveness detection” feladata megállapítani, hogy valóban a hitelesítendő személy látható a képen (bal oldali ábra), vagy kinyomtatott vagy digitális kép (jobb oldali ábra), melyre pl. mélységérzékelő kamerák alkalmazhatók (ábra forrása: [10])

A fenti 8 támadási pont, melyeken keresztül egy támadó jogtalanul hozzáférést adhat vagy tagadhat meg, bármilyen biometrikus azonosítást használó hozzáférés kezelési rendszer esetén fennál. A továbbiakban néhány olyan veszélyt mutatunk be, melyek kifejezetten arcfelismerő rendszerek esetén jelenthetnek problémát, akár hozzáférés kezelési, akár adatvédelemi szempontból.

Kimondottan arcfelismerés esetén egyedi probléma a “face morphing” jelenség, amikor is két vagy több ember arca kerül összevonásra egy fotóba, amelyről a biometrikus sablon készül, oly módon, hogy későbbi azonosítás során mindegyiküket felismerje az arcfelismerő rendszer. Erre mutat példát a Ábra 19. ábra. Ennek a támadásnak a kockázata az, hogy lehetővé teheti, hogy egy banki felhasználói fiókhoz több különböző ember is hozzá tudjon férni, ha a hitelesítéshez használt fénykép vagy biometrikus sablon egy „face morphing” eljárással gyártott képből származik. Ez ellen a támadás ellen kellően robosztus arcfelismerő rendszerekkel lehet védekezni, s fontos, hogy a korrumpált biometrikus sablont még az előtt kiszűrjük, mielőtt eltárolnánk az adatbázisban.



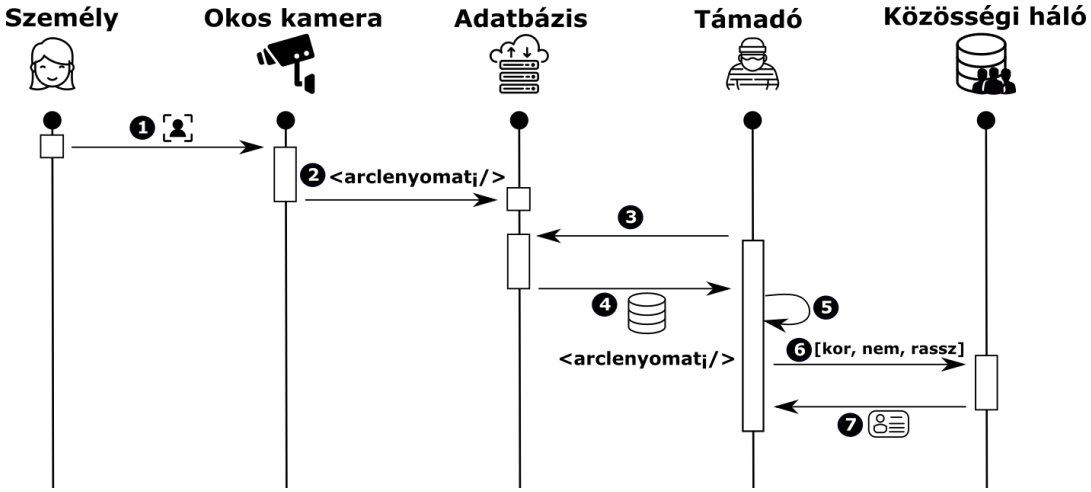
Ábra 19 - A „face morphing” eljárás során két vagy több emberi arcot (bal és jobb oldali kép) egyetlen arcképpé vonnak össze (középső arc), így egy arcfelismerő rendszer tévesen mindkét eredeti arcot hitelesítheti [11]

Egy adatvédelmi kockázatra mutattak rá azok a kutatók, akik bebizonyították, hogy az eddig bemutatott arcfelismerési eljárás inverze is működik, azaz az arcképekből származó arclenyomat vektorokból gépi tanulással az eredeti arcképekhez nagyon hasonló arcképek is visszaállíthatók [8]. Ez adatvédelmi szempontból problémás lehet, ugyanis, ha egy rosszindulatú fél megszerzi a tárolt arclenyomatokat, akkor azokból ezzel az inverz eljárással kideríthetti, hogy melyik arclenyomathoz milyen személyazonosság tartozik, egy banki rendszer esetében tehát például ki tudja deríteni, hogy melyik számlákhoz vagy tranzakciókhoz a bank mely ügyfelei tartoznak.

Ábra 20 – A felső sorban az eredetii arcképek, az alsó sorban pedig az arcképekről készült arclenyomat vektorokból visszaállított arcképek láthatóak. A felső és az alsó kép közötti számok a két kép közötti cosinus hasonlóságot jelölik (ábra forrása: [8]).

Hasonlóan adatvédelmi (újra-azonosítási) problémákat vet fel, hogy megfelelő gépi tanulási eljárásokkal az arclenyomat vektorokból az arclenyomathoz tartozó személy demográfiai adatai is magas pontossággal kikövetkeztethetők [12]. Az ebből adódó egy

lehetséges újra-azonosítási támadásra mutat példát a 18. ábra. Tegyük fel, hogy egy bank a bankfiókban megjelenő ügyfelekről felvételt készít (1. lépés), s a felvételeken megjelenő arcképekből kivont arclenyomat vektorokat egy központi adatbázisban tárolja (2. lépés). Amennyiben egy belső alkalmazott az arclenyomat adatbázist kiszivárogtatja vagy egy külső támadó azt megszerzi (3. és 4. lépés), majd ezekből az arclenyomat vektorokból számos demográfiai adatot kivon, mint például életkor, nem vagy rassz (5. lépés), úgy lehetősége nyílik ezen adatok alapján a bank ügyfeleinek újra azonosítására például egy közösségi hálón való kereséssel (6. és 7. lépés).



Ábra 21 – Emberek újra azonosítása a róluk tárolt arclenyomat vektorokból kinyert demográfiai adatok segítségével. [12]

Összességében elmondható tehát, hogy ugyan az arcfelismerési technológia rendkívül sokat fejlődött az elmúlt években, s pontossága már megközelíti az emberi teljesítményt, azonban a technológia gyakorlati alkalmazása számos kihívással jár, melyek jelenleg is kutatások alapját képezik.

5 Arcfelismerés adatvédelemmel

Az automatikus biometrikus azonosítási eljárások exponenciális terjedése miatt a biometrikus sablonok tárolásával kapcsolatos adatvédelmi aggályok is felerősödtek. A biometrikus sablonok módosítás vagy titkosítás nélküli tárolása ugyanis számos problémát felvet. Például egyes biometrikus jellemzők nyilvános ismerete (pl. arckép) visszaélésekhez vezethet, illetve a jelszavakkal ellentétben egy biológiai tulajdonságot nem lehet könnyen megváltoztatni vagy visszavonni. Mindezekből kifolyólag a biometrikus azonosítási eljárások egyik legfőbb előnye, miszerint az ember biológiai tulajdonságai nem változnak jelentősen az idő múlásával, egyben hátrány is lehet, hiszen ilyen adatok kiszivárgása súlyos adatvédelmi kockázatot jelent.

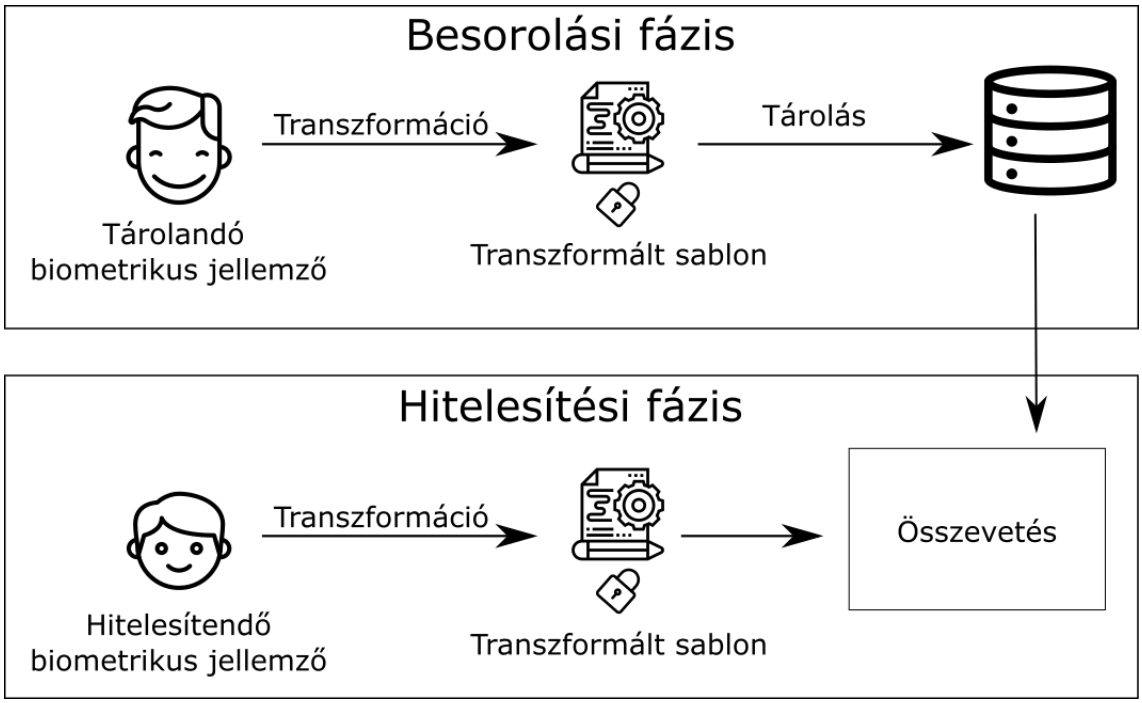
E problémák miatt dolgoztak ki kutatók olyan eljárásokat, melyekkel úgy valósítható meg biometrikus azonosítás, hogy az adatalanyokról készült eredeti biometrikus sablonok tárolására nincs szükség. Ezen eljárások úgynevezett biometrikus sablonvédelemmel valósíthatóak meg, mely a következőkben kerül bemutatásra.

5.1 Alapvető biometrikus sablonvédelem

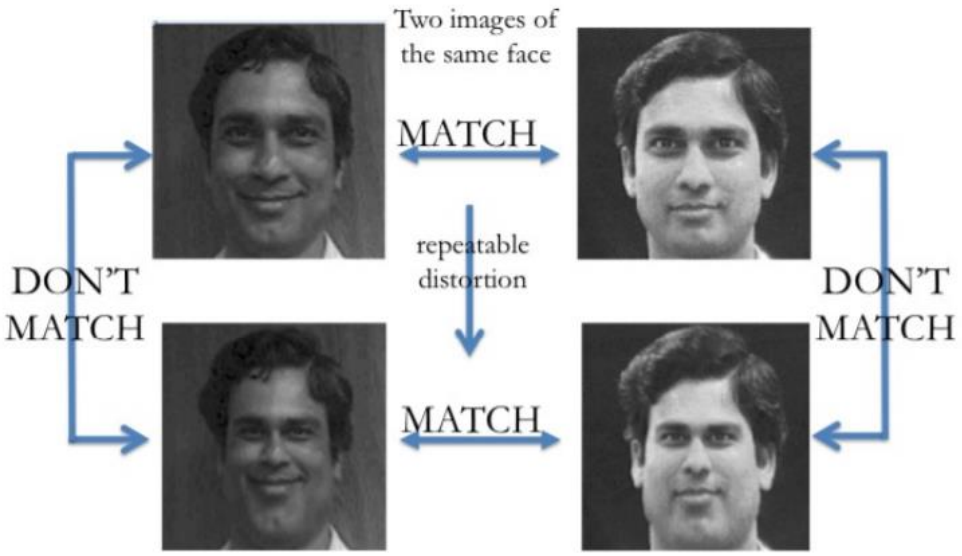
A biometrikus sablonvédelem lényege, hogy az eredeti biometrikus sablonokat (pl. arclenyomat vektor, ujjlenyomat) bizonyos transzformációkkal módosítjuk, így úgynevezett törölhető biometrikus sablonokat kapunk [3]. A törölhető biometrikus sablonokkal történő hitelesítés sematikus bemutatása a Ábra 22. ábrán látható. Alapvetően két fázist különböztetünk meg, a besorolási és a hitelesítési fázist. A besorolás során az adatalany egy biometrikus sablonját egy megfelelő transzformációval átalakítjuk, majd a transzformált sablont eltároljuk. A hitelesítési fázis során a hitelesítendő személy biometrikus jellemzőjét ugyanazzal a transzformációval átalakítjuk, majd az átalakított sablont összevetjük az eltárolt sablonnal, s egyezés esetén történik meg a hitelesítés.

Egy lehetséges transzformációt a Ábra 23 – A geometriai transzformáció egy lehetséges törölhető biometrikus sablon megvalósítási mód, arcfelismerés esetén (az ábra. ábra mutat be, melyen az látható, hogy az eredeti arckép helyett annak geometriai transzformáltja kerül eltárolásra, s egyezés, vagy csak a két eredeti arckép, vagy csak a két transzformált

arckép között mutatható ki, de eredeti és transzformált arckép között nincs egyezés, így a transzformált arckép kiszivárgása esetén nem lehet következtetni az adatalany személyazonosságára.



Ábra 22 - Törölhető biometrikus sablonok alkalmazásának sematikus bemutatása



Ábra 23 – A geometriai transzformáció egy lehetséges törölhető biometrikus sablon megvalósítási mód, arcfelismerés esetén (az ábra forrása: [3])

A törölhető biometrikus sablonok csak bizonyos háttértudás segítségével szolgáltatnak információt az eredeti adatalanyról (ugyanis az eredeti sablon, vagy csak egy visszafejtő

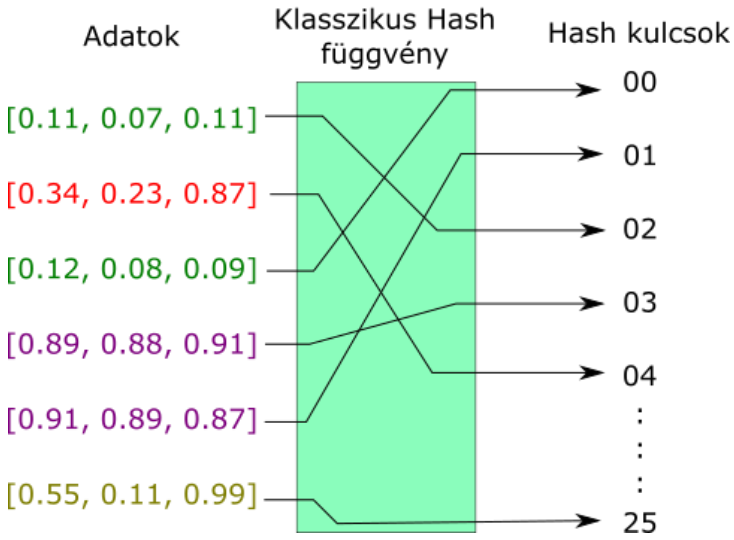
kulcs segítségével, vagy egyáltalán nem állítható vissza), így ezeknek kiszivárgása vagy kompromittálódása nem jelent akkora problémát, mint az eredeti sablonok kiszivárgása. A törölhető biometrikus sablonok megoldják a visszavonhatóság kérdését is, ugyanis lehetőség van egy adott ember eltárolt sablonjának elvetésére, majd egy másik transzformációval új, az előzőtől különböző sablon létrehozására és eltárolására.

A biometrikus sablonvédelem megvalósítására használhatunk titkosítási vagy hashelési eljárásokat is. Titkosításnak azt az eljárást nevezzük, amikor egy szöveget egy titkosító algoritmus és egy kulcs segítségével olyan titkosított szöveggé alakítunk, mely csak olyan ember számára lesz értelmezhető, aki rendelkezik az olvasáshoz szükséges kulccsal. A hashelés ezzel szemben olyan eljárás, melynek során bármilyen hosszúságú szöveget egy adott hosszúságú adatblokkra (pl. karakterláncra) képezünk le. A lényegi különbség a két eljárás között az, hogy míg a titkosítás során visszanyerhető az eredeti szöveg (a megfelelő kulcs ismeretében), addig a hashelés egyirányú, nem invertálható függvény, tehát a hashelés eredményeképp keletkező hash-ből nem állítható egyértelműen vissza az eredeti szöveg. Mivel a keletkező hash minden esetben fix, véges számú karakterből áll, ezért véges a lehetséges hash értékek száma, így a végtelen lehetséges bemenet (az összes elképzelhető szöveg) véges számú hash érték egyikére képződhet csak le. Ebből következik, hogy akár eltérő tartalom is eredményezhet azonos hash kulcsot, amit ütközésnek nevezünk. Azonban minél nagyobb a választott hash kulcs mérete, annál kisebb az ütközések valószínűsége.

Titkosítással úgy valósítható meg biometrikus sablonvédelem, hogy csak a titkosított sablont tároljuk, s csak arra az időre fejtjük vissza az eredeti sablont, ameddig összevetjük a hitelesítendő sablonnal. Hashelés során pedig azzal a feltételezéssel élünk, hogy ideális esetben az azonos emberről készült biometrikus sablonok hash kulcsai megegyeznének (ütköznek), így elég csak az eredeti sablon hash kulcsát tárolni, s ütközés esetén történik meg a hitelesítés.

Bár kézenfekvőnek tűnhet klasszikus titkosítási vagy hashelési eljárások alkalmazása biometrikus sablonok védelme esetén is, azonban ez legtöbbször nem járható út. A klasszikus titkosítási eljárásokkal az a probléma, hogy az adatok felhasználásához mindenképpen szükséges azok visszafejtése (ugyanis a sablonok csak visszafejtett, titkosítatlan állapotban hasonlíthatóak össze), s ez továbbra is adatvédelmi kockázatot jelent. A hashelés esetén pedig az jelent problémát, hogy a klasszikus hashelési eljárások

célja az ütközések minimalizálása, ami miatt érzékenyek a bemenet változásaira, s a legkisebb eltérés hatására is teljesen különböző hash kulcsokat generálnak, ahogy azt a Ábra 24. ábra szemlélteti (ezt hívjuk lavinahatásnak). S mivel még az ugyanarról az emberről készült biometrikus sablonok sem egyeznek meg teljes mértékben szinte soha, így a klasszikus hashelés alkalmazása nagyon magas arányú hamis elutasítást eredményezne [3]. Ezen problémák miatt biometrikus sablonvédelemhez speciális titkosítási és hashelési eljárásokat kell alkalmazni, melyekről a későbbiekben lesz szó.



Ábra 24 – Klasszikus hashelés működése: az ütközések minimalizálása miatt még a hasonló, egy csoportba tartozó (azonos színnel jelölt) adatok is különböző hash kulcsot kapnak.

Az utóbbi években megjelentek olyan biometrikus sablon védelmi eljárások is, melyek nem titkosításra vagy hashelésre alapulnak. Például az egyik ilyen megoldás egy vizuális kriptográfiai eljárás, mely során az adatalanyról készült eredeti képet szétbontják két zajos képre, és ezt a két képet két különböző adatbázisban tárolják el [3]. A hitelesítés során a hitelesítendő képet aztán a két zajos kép együtteséből visszaállított eredeti képpel vetik össze. Az eljárás előnye, hogy az egyik adatbázis kiszivárgása esetén nem lehet az adatalanyok képeit helyreállítani, hiszen ahhoz mindkét adatbázisra szükség lenne. Hátránya is ugyanebből ered, ugyanis két adatbázis kezelése bonyolultabbá teszi az alkalmazást és nem minden esetben praktikus.

Egy másik példa hashelés és titkosítás nélküli biometrikus sablonvédelemre az az eljárás, mely során nem egy ujjlenyomatot tárolnak el az adatalanyról, hanem két különböző ujjáról vett ujjlenyomat felhasználásával alkotnak egy új sablont, s ez a sablon kerül csak eltárolása [3]. Ez esetben az adatbázis kiszivárgása esetén nem kompromittálódnak az

adatalanyok ujjlenyomatai, hiszen a tárolt összevont ujjlenyomat sablonokból az eredeti egyedi ujjlenyomatok nem állíthatók vissza.

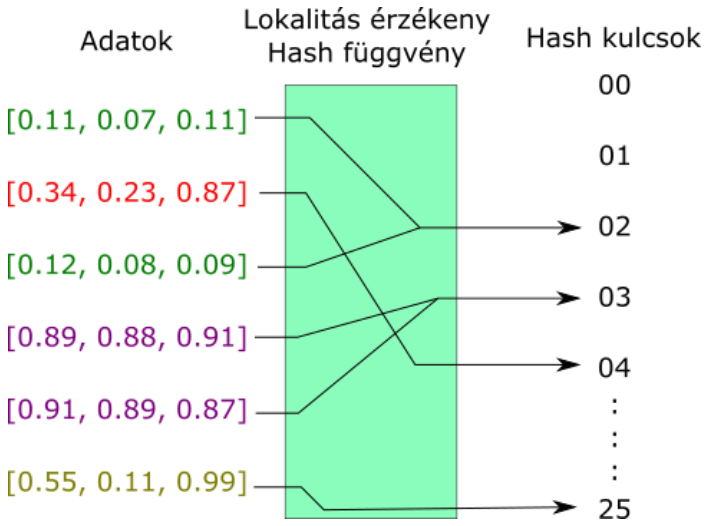
Összefoglalva tehát a biometrikus sablonvédelmi eljárások a biometrikus azonosító rendszerek egyik legkomolyabb adatvédelmi kockázatát mérséklik, az által, hogy a „nyers" biometrikus sablonok kiszivárgását, s ez által érzékeny adatok idegen kezekbe jutását akadályozzák meg. Fontos azonban megjegyezni, hogy amennyiben a sablon transzformációk kulcsa kiszivárog, és a transzformáció invertábilis (pl. titkosítás esetén), akkor a sablonok a kulcs ismeretében visszaállíthatóak. Sőt, ritka esetekben még nem invertábilis transzformációk esetén is előfordulhat, hogy a támadó kellő pontossággal vissza tudja állítani az eredeti biometrikus sablont [3]. Mindezek miatt tehát fontos hangsúlyozni, hogy a biometrikus sablonvédelem mellett is komoly hangsúlyt kell fordítani az adatbiztonságra.

5.2 Véletlen projekció, mint hashelés

Az előzőkben említett problémák miatt biometrikus sablonok hashelésére klasszikus hashelési eljárások az ütközés-minimalizációjuk miatt nem alkalmazhatók. Ez a probléma megoldható az ún. „hely érzékeny" hash függvények használatával, amelyek sokkal kevésbé zajérzékenyek, s így az egymáshoz hasonló biometrikus sablonokat szintén egymáshoz hasonló hash kulcsokká képeznek le.

5.2.1 Elméleti háttér

A Ábra 25. ábrán látható, hogy mennyiben más az úgynevezett lokalitás érzékeny hash függvények működése a klasszikus hash függvényekhez képest. Mint ahogy az ábrán látható, az azonos színnel jelölt, azonos csoportba tartozó, de nem teljesen megegyező adatok azonos hash kulcsot kapnak.



Ábra 25 - A lokalitás érzékeny hash függvények működése: az ütközések maximalizálása miatt a hasonló adatok azonos hash kulcsot kapnak, még úgy is, hogy nem egyeznek meg teljesen.

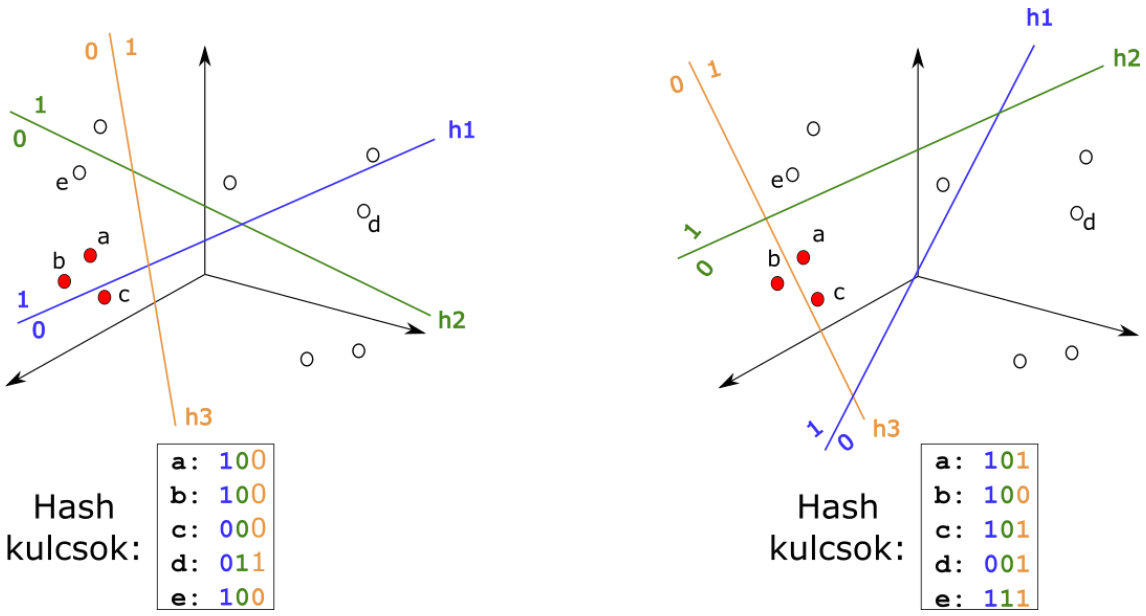
A lokalitás érzékeny hash függvények létezését a Johnson–Lindenstrauss lemma mondja ki, mely szerint egy magas dimenziójú térben elhelyezkedő pontok halmaza leképezhető egy alacsonyabb dimenziójú térben elhelyezkedő pontok halmazára oly módon, hogy bármely két pont közötti relatív távolság megmaradjon [3]. Ebből az következik, hogy az eredeti magas dimenziójú adatpontokat le lehet képezni olyan alacsonyabb dimenziójú adatpontokra, hogy az eredeti adatpontok statisztikai jellemzői megmaradjanak.

Az ilyen távolságtartó hashelési eljárásokat angolul Locality Sensitive Hashing-nek nevezzük (LSH). Az *LSH* egy gyűjtő fogalom, amely olyan hash függvényeket foglal magába, amik a Johnson-Lindenstrauss lemma értelmében adatok csoportosítására használhatók, hiszen hasonló bemeneteket hasonló csoportokba képeznek le. Ezzel a módszerrel akár biometrikus sablonok is csoportosíthatók anélkül, hogy az eredeti biometrikus sablont tárolni kéne, hiszen elég csak a sablonok hash kulcsait tárolni.

Erre mutat példát a Ábra 26. ábra, ahol különböző pontokat szeretnénk csoportosítani, oly módon, hogy az egymáshoz közel eső pontok hash kulcsai megegyezzenek (pl. a három pirossal jelölt a, b, c pont hash kulcsai megegyezzenek). Ehhez a teret felosztjuk három véletlenszerűen választott h_1, h_2, h_3 hipersíkkal, s minden pont hash kulcsát az határozza meg, hogy melyik sík melyik oldalára kerül. Azon pontok, melyek minden síknak ugyanazon az oldalán helyezkednek el, ugyanolyan hash kulcsot kapnak, így azok tekinthetők egy csoportba tartozónak. Mint az ábrán látható, attól

függően, hogy a három hipersíkot hogyan választjuk meg, más és más lehetséges hash kulcsok állhatnak elő.

Például az ábra bal oldalán egy olyan eset látható, amikor az a, b és e pontok hash kulcsai megegyeznek, ugyanakkor a c pont hash kulcsa különböző. Ez azt eredményezheti, hogy az e pont tévesen azonos csoportba kerül az a, b pontokkal, míg a c pont tévesen másik csoportba sorolódik. Ennek elkerülésére a hashelést általában nem egyszer, hanem többször, több véletlenszerűen választott hipersíkkal ismételjük meg, s azon pontokat, melyek legalább egy esetben azonos hash kulcsot kapnak, azonos csoportba tartozónak tekintjük. Így az ábra jobb oldalán látható, hogy már a c pont is helyesen egy csoportba kerül az a ponttal, tehát ez az eljárás segít megtalálni az azonos csoportba tartozó pontokat. A hash kulcs méretének növelésével (több hipersík használata) pedig csökkenthetjük annak az esélyét, hogy olyan pontok azonos hash kulcsot kapjanak, amelyek valójában nem tartoznak egy csoportba.



Ábra 26 – Az LSH egy lehetséges implementációja

Egy, az LSH családba tartozó módszer a véletlen vetítés módszer (*random projection method*) [3]. A módszer lényege, hogy az egyes adatpontok dimenziója csökkenthető úgy, hogy az N dimenziós x_i adatpontokat egy $n < N$ dimenziós altérben elhelyezkedő, y_i pontokra vetítjük le, a pontok közötti relatív távolságok jelentős megváltozása nélkül. A vetítéshez egy A mátrixot kell generálnunk, amelynek elemei

Gauss-eloszlásúak, az átlaguk 0, a varianciájuk pedig 1, s ekkor az y_i pontokat úgy kapjuk, ha az x_i pontokat megszorozzuk az A mátrixszal:

$$y_i = A \cdot x_i$$

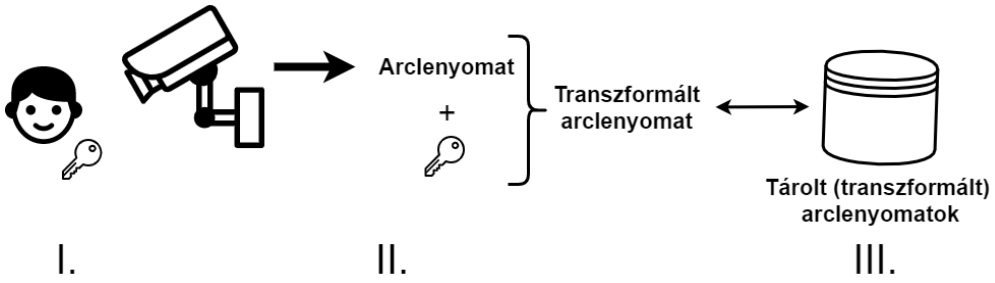
ahol:

- y_i – az alacsonyabb dimenziójú térben elhelyezkedő pontok (pl. a törölhető biometrikus sablonok)
- A – a vetítéshez használt mátrix
- x_i – a magasabb dimenziójú térben elhelyezkedő pontok (pl. az eredeti biometrikus sablonok)

5.2.2 Véletlen vetítéses módszer alkalmazása arcfelismeréshez

A véletlen vetítéses módszerrel alapvetően kétféleképpen valósítható meg biometrikus sablonvédelemmel ellátott arcfelismerő rendszer.

Egyrészt lehetséges (lásd 24. ábra), hogy a rendszer a felhasználókról csak a transzformált biometrikus sablonokat (tehát a véletlen vetítéssel redukált dimenziójú arclenyomatokat) tárolja (azaz csak y_i-ket), se az eredeti sablonokat, se a transzformációs mátrixot nem. Ebben az esetben a felhasználónak kell hitelesítés során a transzformációs mátrixot is szolgáltatni (pl. egy smart kártya segítségével), s amennyiben a róla készült biometrikus sablont (vagyis az arcképéből kinyert arclenyomatot) az általa szolgáltatott mátrix olyan pontba vetíti, mely közel van az eltárolt sablonhoz, akkor a felhasználó hitelesítése megtörténik. Ebben az esetben a tárolt sablonok esetleges kiszivárgása sem jelent veszélyt, hiszen csak a transzformált sablonok kerülnek tárolásra, illetve a felhasználóknál tárolt transzformációs mátrixok egy plusz biztonsági réteget szolgáltatnak.



Ábra 27 – A véletlen vetítés egyik lehetséges alkalmazása arcfelismerésre. Az I. fázisban a felhasználó megjelenik a kamera előtt, s szolgáltatja a transzformációs kulcsot is (vetítési mátrix, ami pl. egy felhasználó által hordott smart kártyán van tárolva). A II. fázisban a kamera által készített arcképből kivonásra kerül az arclenyomat vektor, majd azt a kulccsal transzformálja a rendszer. A III. fázisban a transzformált arclenyomat vektort a rendszer összeveti az eltárolt (szintén transzformált) arclenyomat vektorokkal, egyezést keresve. Módosítatlan arclenyomat vektorok soha nem kerülnek tárolásra, illetve a transzformációs kulcsot is a felhasználó szolgáltatja, így növelve a rendszer biztonságát.

A másik esetben a transzformált sablon mellett a transzformációs mátrixot is eltárolja a rendszer. Ebben az esetben a hitelesítés során kinyerik a felhasználó arcképéből az arclenyomat vektort, majd az eltárolt transzformációs mátrix-szal azt transzformálják, s amennyiben az így keletkező transzformált sablon közel van az eltárolt sablonhoz, akkor történik meg a hitelesítés. Ez az eset ugyan kényelmesebb felhasználói szempontból, hiszen a felhasználónak nem kell magával hordoznia a transzformációs mátrixot (pl. a smart kártyát), ugyanakkor kevésbé biztonságos, hiszen egy támadó mind a mátrixot, mind a transzformált biometrikus sablonokat megszerezheti egy adatszivárgás során. Ugyanakkor mivel ez esetben is transzformált sablonok kerülnek csak tárolásra, ezért ez a rendszer is biztonságosabb az eredeti sablonokat tároló megoldásoknál.

A véletlen vetítéses módszernek több előnye van. Egyrészt még a transzformált sablonok és a transzformációs kulcs kiszivárgása esetén sem lehetséges az eredeti sablonok visszaállítása a vetítés miatt bekövetkező dimenzió csökkenés és a transzformáció nem-invertálhatósága miatt [3]. Másrészt amennyiben a támadó megszerzi valahogy egy felhasználó eredeti biometrikus jellemzőjét (pl. arcképét), a transzformációs kulcs ismerete nélkül továbbra sem hitelesíti a rendszer (amennyiben a felhasználónak kell a kulcsot is szolgáltatni, nem pedig a rendszer tárolja). További előny, hogy amennyiben bármelyik eltárolt adat kompromittálódik, akkor lehetséges új transzformációs mátrix létrehozása, mellyel új transzformált sablonok gyárthatóak, s így lehetőség van az adatbázisban tárolt adatok cseréjére, vagyis teljesül a visszavonhatóság.

5.3 Kriptográfia alkalmazása

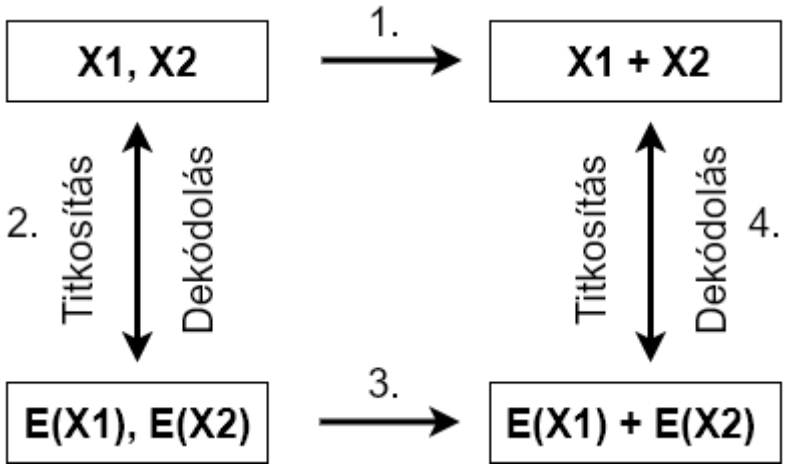
Bár a klasszikusnak számító kriptográfiai titkosítási módszerek praktikusan feltörhetetlennek számítanak, azonban biometrikus sablonok kezelésére mégsem ideális az alkalmazásuk. Ennek oka az, hogy műveletek nem végezhetők titkosított (kódolt) adatok között, ezért műveletek végzése előtt az adatok visszafejtése (dekódolása) szükséges. Tehát amennyiben két biometrikus sablont össze akarunk hasonlítani, például két arclenyomat vektor közötti Euklidészi távolságot szeretnénk meghatározni, akkor a titkosított arclenyomat vektorokat előbb dekódolni szükséges, s ez adatvédelmi kockázatot rejt, hiszen ezen a ponton kiszivároghatnak a titkosítatlan arclenyomat vektorok.

A biometrikus sablonok védelméhez tehát olyan titkosítási eljárásra van szükség, mely lehetővé teszi a titkosított állapotban való alkalmazást a visszafejtés szükségessége nélkül. Így az eredeti biometrikus sablonokat soha nem kell eltárolni, csak a titkosított változatukat, s az alkalmazás működése során soha nem kell titkosítatlan sablonokat kezelni. Egy ilyen lehetséges titkosítás a homomorfikus titkosítás, melyről az alábbiakban lesz szó.

5.3.1 Homomorfikus titkosítás elméleti háttere

A homomorfikus titkosítás megoldást nyújthat a fenti igényekre. A homomorfikus titkosítás az aszimmetrikus kulcsú titkosítás egy speciális fajtája. Aszimmetrikus titkosítás (másik nevén nyilvános kulcsú rejtjelezés) során két különböző kulcsot használunk: egy publikus kulcsot az adatok titkosítására, s egy privát kulcsot a titkosított adatok visszafejtésére. Az aszimmetrikus kulcsú titkosítás lényege, hogy a publikus kulccsal mindenki rendelkezhet, így bárki tud olyan titkosított üzentet küldeni, melyet csak a címzett lesz képes dekriptálni a privát kulcs birtokában [37].

A homomorfikus titkosítás az aszimmetrikus titkosítás egy olyan implementációja, amely lehetővé teszi a titkosított adatokon történő műveletek végzését. A titkosított adatokon végzett műveletek eredménye szintén egy titkosított adat, mely dekódolva ugyanazt az eredményt adja, amit a titkosítatlan adatokon elvégzett ugyanezen műveletek eredményeztek volna [38] (lásd Ábra 28. ábra).



Ábra 28 - A homomorfikus titkosítás működése. Az 1. és 3. lépés egy adott művelet elvégzését jelenti (pl. összeadás), a 2. és 4. lépés pedig a titkosítást/dekódolást. Az eredeti adatokon (X1, X2) elvégzett művelet eredménye megegyezik a titkosított adatokon (E(X1), E(X2)) elvégzett művelet eredményének dekódoltjával, tehát az 1. lépés eredménye elérhető a 2-3-4. lépésekkel is.

Előnye, hogy elég a titkosított adatokat megosztani harmadik féllel (pl. felhő szolgáltatóval), aki tud rajta műveleteket végezni, s mind az eredeti adatokat, mind a műveletek eredményét csak a tulajdonos ismeri a privát kulcs birtokában. Emiatt alkalmazása preferált olyan helyzetekben, amikor például harmadik fél számára szükséges érzékeny adatokat szolgáltatni (pl. egészségügyi vagy banki adatok felhő alapú szolgáltató által történő tárolása vagy feldolgozása).

A homomorfikus titkosításnak három különböző szintű gyakorlati megvalósítása létezik [38]:

- Somewhat homomorphic encryption (SHE, majdnem homomorfikus titkosítás): csak korlátozott számú alkalommal ismételhetők meg műveletek a titkosított adatokon.
- Partially homomorphic encryption (PHE, részlegesen homomorfikus titkosítás): ugyan korlátlan számú alkalommal ismételhetőek meg a műveletek a titkosított adatokon, azonban nem minden művelet támogatott (pl. vagy csak szorzás, vagy csak összeadás).
- Fully homomorphic encryption (FHE, teljesen homomorfikus titkosítás): minden művelet korlátlan számú alkalommal ismételhető meg a titkosított adatokon.

Bár a PHE és FHE esetén is korlátlan alkalommal ismételhetők meg a műveletek, azonban előfordulhat, hogy a titkosítás miatt megnő a zaj mértéke a nyílt adatban. Ezért ezeket a sémákat időnként frissíteni kell, hogy a zaj mértéke lecsökkenjen.

A homomorfikus titkosításnak számos adatvédelmi szempontból előnyös alkalmazása van, de nem hátrányok nélkül: az egyik jelentősebb korlátja a magas számítási igény, mely nem teszi ideálissá használatát gyors működést igénylő alkalmazások esetén. A leglassabb és legszámításigényesebb az FHE titkosítási séma, ezért amennyiben egy alkalmazásban nincs feltétlenül szükség mind az összeadás mind a szorzás műveletekre, úgy célszerű valamilyen gyorsabb PHE titkosítási sémát alkalmazni.

5.3.2 Poszt-kvantum biztonság

Titkosítási sémák esetén fontos azok poszt-kvantum biztonságosságának vizsgálata is, azaz annak eldöntése, hogy az adott titkosítási séma feltörésére milyen veszélyt jelenthet a kvantum számítógépek fejlesztése. A kvantumszámítógépek olyan eszközök, melyek olyan kvantummechanikai jelenségeket használnak, mint a kvantum-szuperpozíció és a kvantum-összefonódás, melyekkel képesek lehetnek a hagyományos számítógépekkel rendkívül időigényes számítások gyors és hatékony elvégzésére [43]. Bár a kvantum számítógépek fejlettsége ma még nem éri el azt a szintet, hogy veszélyt jelentsenek a ma széleskörűen alkalmazott titkosítási sémákra, azonban a jövőben elképzelhető, hogy ez megváltozik.

A ma alkalmazott legnépszerűbb titkosítási sémák mindegyike valamilyen matematikai „nehéz feladaton” (hard problem) alapszik, mint például a prímfaktorizálás, azaz számok prím szorzóira bontása (pl. 100=2·2·5·5). Bár ennek kiszámítása kellően nagy számok esetén hagyományos számítógépekkel akár több száz vagy több ezer évbe is telhet, léteznek olyan kvantumszámítógépre tervezett eljárások (pl. Shor-algoritmus [42]), melyekkel nagyon gyorsan és hatékonyan megoldható ez a feladat.

Bár vannak olyan FHE és PHE titkosítási sémák, melyek az alapjukul szolgáló nehéz matematikai feladat következtében poszt-kvantum is biztonságosnak tekinthetőek, ez nem mondható el minden esetben. Például a fentebb említett Paillier PHE titkosítási séma alapjául az ún. kvadratikus reciprocitás tétele szolgál, amely nem tekinthető biztonságosnak kvantum számítógépekkel szemben [41]. Ugyanakkor kutatók

folyamatosan dolgoznak poszt-kvantum FHE és PHE titkosítási sémák megalkotásán. Példaként említhető két 2020-ban publikált additív PHE titkosítási eljárás, az FAHE1 és FAHE2 (Fast Additive Homomorphic Encryption) [41], vagy a rácselméleti háttérre építő NTRU-n (Nth Degree Truncated Polynomial Ring Units) alapuló FHE sémák [38], melyekről jelenleg úgy gondolják, hogy kvantum számítógépekkel szemben is ellenállóak lehetnek.

Összegzésül elmondható, hogy a homomorfikus titkosítási sémák fejlesztése jelenleg is aktívan kutatott terület, így a jövőben egyre gyorsabb, hatékonyabb és kvantum számítógépekkel szemben is biztonságosnak minősülő titkosítási eljárások megjelenése várható. Ebből kifolyólag a homomorfikus titkosítás egy nagyon ígéretes megoldás lehet a jövőben a privacy-preserving biometrikus azonosítás alkalmazása során.

5.3.3 Homomorfikus titkosítás alkalmazása arcfelismeréshez

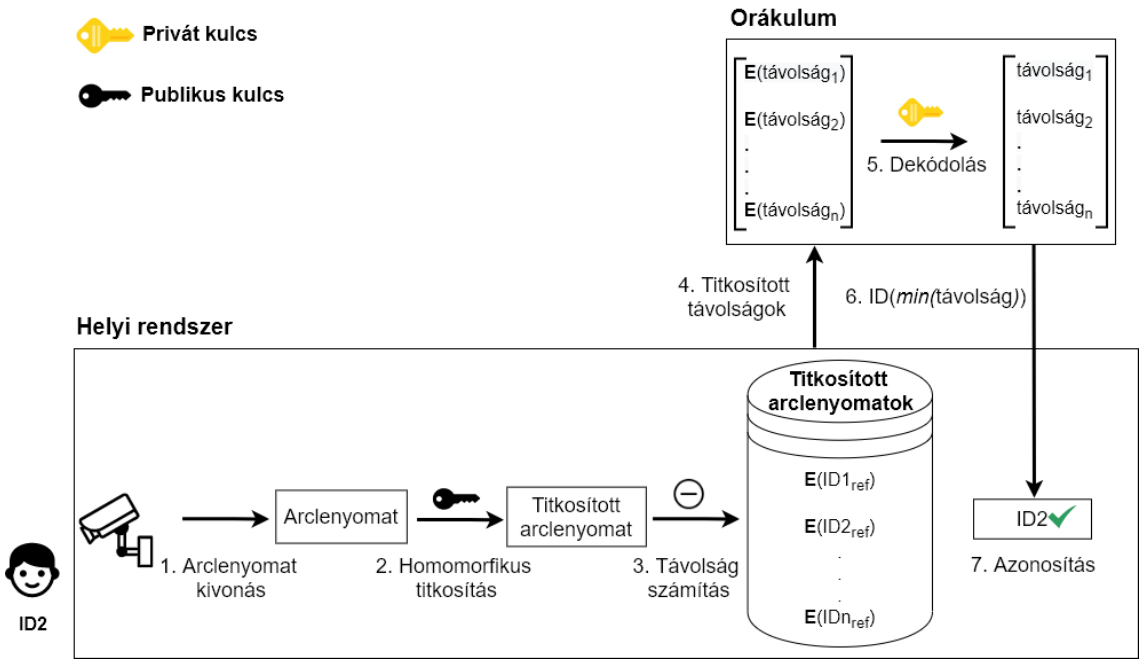
Biometrikus sablonok esetén sokszor elegendő lehet PHE titkosítási séma alkalmazása is. Például arcfelismerés esetén azonosításhoz mindössze arra van szükség, hogy arclenyomat vektorok távolságát meg tudjuk határozni, melyhez pl. Manhattan távolság esetén elegendő az összeadás művelet támogatása a titkosított arclenyomatokon, s erre például alkalmas a Pailler PHE séma [39].

A Pailler PHE séma alkalmazására mutat egy lehetséges megoldást a Ábra 29. ábra. Új felhasználó felvétele esetén a felhasználóról készült referencia arclenyomat vektor (IDn_{ref}) először a homomorfikus titkosítás publikus kulcsával titkosításra kerül, majd az így titkosított arclenyomat vektor ($\boldsymbol{E(IDn_{ref})}$) kerül eltárolásra a helyi rendszer adatbázisban, az eredeti arclenyomat vektort elvetjük. Minden felhasználó felvétele esetén ezt az eljárást alkalmazzuk. A felismerés pedig a következőképpen történik.

Amikor megjelenik valaki a kamera előtt, a kamera által készített arcképből kivonásra kerül az illető arclenyomata (1. lépés). Az arclenyomatot ezután a publikus kulccsal titkosítjuk (2. lépés), az eredeti arclenyomatot elvetjük, s titkosított formában összevetjük az eltárolt titkosított referencia arclenyomat vektorokkal (3. lépés). Mivel a Pailler PHE séma az összeadást támogatja, ezért az újonnan titkosított arclenyomat vektor és a tárolt titkosított arclenyomatok Manhattan távolsága kiszámítható. Mivel a kiszámított eredmények továbbra is titkosítottak, így a privát kulcsra van szükség ahhoz,

hogy a megkapjuk a konkrét Manhattan távolságokat az új arclenyomat, illetve a tárolt arclenyomatok között.

Ebben a sémában a titkosítás feloldásáért egy ún. „orákulum” felel, mely egy olyan megbízható harmadik fél, akinek birtokában van a privát kulcs. A helyi rendszer a titkosított távolságokat tovább küldi az orákulumnak (4. lépés), aki aztán a privát kulccsal feloldja a titkosítást (5. lépés), s megállapítja, hogy melyik referencia arclenyomattal vett távolság a legkisebb. Végül az orákulum csak azt küldi vissza, hogy melyik személyazonossággal volt a legkisebb a távolság (6. lépés), további információt nem közöl a helyi rendszerrel. Így elérhető, hogy a helyi rendszer sikeresen azonosítja a kamera előtt megjelenő személyt (7. lépés), azonban semmilyen egyéb információhoz nem jut, hiszen csak titkosított arclenyomatokat tárol, s a privát kulcs hiányában nem tudja a titkosítást feloldani.



Ábra 29 – A homomorfikus titkosítás alkalmazása arcfelismerésre egy olyan rendszerben, melyben a helyi szerver egy orákulummal kommunikál. A helyi szerver csak titkosított arclenyomatokat tárol, s rendelkezik a publikus kulccsal, melyek további arclenyomatokat tud titkosítani. A titkosítás feloldásához szükséges privát kulccsal azonban csak a távoli szerver, avagy orákulum rendelkezik, amelyik sosem érintkezik a titkosított arclenyomatokkal, csak azok különbségével, így növelve a biztonságot.

Ily módon lehetőség nyílik arra, hogy csak Pailler sémával titkosított arclenyomat vektorokat tároljuk az adatbázisban, s hitelesítés során is csak titkosított arclenyomat vektorokat hasonlítsunk össze. Az egyetlen pont, ahol titkosított adatok dekódolásra kerülnek, az az, amikor a Manhattan távolságokat dekódolja az orákulum, de ezekből semmi nem derül ki az eredeti arclenyomat vektorokat illetően, így ez nem jelent adatvédelmi kockázatot. Így amennyiben csak a helyi szerver által tárolt titkosított arclenyomatok, vagy csak az orákulum privát kulcsa szivárog ki, nem kerülnek idegen kezekbe a felhasználók biometrikus sablonjai.

6 Összefoglalás

Tanulmányunkban bemutattuk az egyre szélesebb körben elterjedő biometrikus azonosítási rendszerek csoportosítását, működését és főbb jellemzőit, valamint kitértünk a biometrikus rendszerek alkalmazásával járó lehetséges előnyökre és hátrányokra. Kifejtésre kerültek a biometrikus rendszerek alkalmazásával járó adatvédelmi kockázatok is, s az európai általános adatvédelmi rendelet (GDPR) biometrikus adatokra vonatkozó szabályozása. Továbbá ismertettük az arcfelismerés, mint az egyik lehetséges biometrikus azonosítási módszer, alkalmazási területeit, mind az állami és magán szektorban.

Külön fejezetben tárgyaltuk az arcfelismerési eljárások működését, melyben kitértünk az arcdetektálás és az arcfelismerés részleteire is. Bemutattuk napjaink legelterjedtebb mély tanulásra épülő arcfelismerési megoldásainak alapelveit, beleértve a deep metric learning, a távolság metrikák, a triplet loss és a sziámi hálók működését. A ma létező legkorszerűbb arcfelismerési rendszerek is bemutatásra kerültek, beleértve a privát (DeepFace, FaceNet) és a szabadon hozzáférhető (OpenCV, dlib, InsightFace) programozási könyvtárakat. Minden könyvtár esetén bemutatásra került a könyvtár által használt arcdetektor működése (kitérve az egyes arcdetektálási technikák működésére), illetve az arcfelismeréshez használt neurális háló architektúrája, tanítási módja, a tanításhoz használt adathalmaz, illetve a referenciaként használt adathalmazon elért pontossága is.

A továbbiakban részletesen kifejtettük az arcfelismerés során keletkező arclenyomat vektorok (más néven embeddingek) azonosításra és hitelesítésre történő felhasználását klaszterezéssel, illetve küszöb sémával. Emellett az osztályozás alapú hitelesítés is bemutatásra került, kitérve több különböző gépi tanulási osztályozási technikára is, mint például a döntési fa, random forest, support vector machine, vagy neurális háló, melyek mind alkalmazhatók az emberek arclenyomat vektoraik alapján történő osztályozására.

Tanulmányunkban az arcfelismerés gyakorlati alkalmazásának nehézségeire és kockázataira is kitértünk. Írtunk a mély tanulásos rendszerekbe a tanító adathalmazokon keresztül bekerülő részrehajlások által jelentett kockázatokról (pl. afro amerikai és ázsiai

rasszba tartozó emberek esetén magasabb hamis felismerési arány), a tömeges arcfelismerés által jelentett társadalmi kockázatokról, illetve az arclenyomat vektorok tárolásával kapcsolatos adatvédelmi problémákról. Elmondható, hogy arclenyomat vektorok (és más biometrikus sablonok) tárolása komolyabb adatvédelmi kihívásokat jelent, mint jelszavak vagy PIN kódok tárolása, hiszen az emberek biometrikus tulajdonságai nem megváltoztathatóak vagy visszavonhatóak. Ezen kívül bemutattuk a biometrikus azonosításra épülő hozzáférés kezelő rendszerek lehetséges támadási pontjait az ellenük bevethető védekezési lehetőségekkel, s részleteztük a kifejezetten arcfelismerést használó rendszerekre veszélyt jelentő támadási módszereket is (face morphing, arc visszaállítás arclenyomatokból, demográfiai adatok kiszivárgása arclenyomatokból).

Az utolsó fejezetben bemutattuk a törölhető biometrikus sablonok alkalmazásának lehetőségeit, melyek segítségével privacy preserving módon lehet biometrikus azonosítást és hitelesítést megvalósítani. A bemutatott, locality sensitive hashelésre épülő technika a véletlen projekció, mely segítségével úgy módosíthatóak a biometrikus sablonok, hogy a módosított sablonokból az eredeti sablonok visszaállítása nem lehetséges, viszont azok azonosításra továbbra is alkalmazhatók, s visszavonhatók, elvethetőek, akár a jelszavak. Egy másik bemutatott technika a homomorfikus titkosítás alkalmazása, mely egy olyan titkosítási technika, ami lehetővé teszi a titkosított adatokon történő műveletek végzését, így elkerülhetővé téve az érzékeny adatnak minősülő arclenyomatok dekriptálását az azonosítás során.

Köszönetnyilvánítás

A tanulmány a Magyar Nemzeti Bank és a Budapesti Műszaki és Gazdaságtudományi Egyetem között létrejött Együttműködés keretében és finanszírozásával készült a Digitalizáció, mesterséges intelligencia és adatkorszak műhelyben.

Az ábrákhoz használt ikonokat a FlatIcon.com weboldalról *eucalyp, freepik, gregor cresnar, pixel perfect, prettycons, surang, becris, dDara, wanicon, xnimrodx* és *monkik* felhasználók szolgáltatták.

Irodalomjegyzék

[1] *GDPR: Adatvédelem mindenkinek - Biometrikus azonosítással kapcsolatos félreértések*, 2020, webcím: https://gdpr.blog.hu/2020/07/10/biometrikus_azonositassal_kapcsolatos_felreertesek

[2] Thales Group, *Biometrics: definition, trends, use cases, laws and latest news*, 2021, webcím: https://www.thalesgroup.com/en/markets/digital-identity-and-security/government/inspired/biometrics

[3] Vishal M. Patel, Nalini K. Ratha, Rama Chellappa, *Cancelable Biometrics: A Review*, 2018, webcím: https://engineering.jhu.edu/vpatel36/wp-content/uploads/2018/08/SPM_CB_v6.pdf

[4] Rahul Kumar, *Support Vector Machine*, 2018, webcím: https://medium.com/@rahulboy002/support-vector-machine-87cafcd0acdc

[5] Wikipedia, *Mesterséges neurális hálózat*, webcím: https://hu.wikipedia.org/wiki/Mesters%C3%A9ges_neur%C3%A1lis_h%C3%A1l%C3%B3zat

[6] Adrian Rosebrock, Siamese networks with Keras, TensorFlow, and Deep Learning, 2020, https://www.pyimagesearch.com/2020/11/30/siamese-networks-with-keras-tensorflow-and-deep-learning/

[7] Office of the Privacy Commissioner of Canada , Automated Facial Recognition In the Public and Private Sectors, https://www.priv.gc.ca/media/1765/fr_201303_e.pdf

[8] Guangcan Mai, Kai Cao, Pong C. Yuen, Anil K. Jain, *On the Reconstruction of Face Images from Deep Face Templates*, 2018, https://ieeexplore.ieee.org/document/8338413

[9] Sweeney, L. (2000). Simple demographics often identify people uniquely. *Health (San Francisco), 671*, 1-34.

[10] Brian Rhodes, Charles Rollet, *Facial Recognition Systems Fail Simple Liveness Detection Test*, 2019, https://ipvm.com/reports/live-detect

[11] Ulrich Scherhag, Luca Debiasi, Christian Rathgeb, Christoph Busch, Andreas Uhl, *Detection of Face Morphing Attacks Based on PRNU Analysis*, 2019,

https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8846232

[12] István Fábián, Gábor Gulyás, *De-anonymizing Facial Recognition Embeddings*, 2020, Infocommunications Journal, https://www.infocommunications.hu/documents/169298/4665731/InfocomJ_2020_2_7_Fabian.pdf

[13] OpenFace, https://cmusatyalab.github.io/openface/

[14] dlib Python könyvtár, https://github.com/davisking/dlib

[15] openCV Python könyvtár, https://opencv.org/

[16] InsightFace Python könyvtár, https://github.com/deepinsight/insightface

[17] face_recognition Python könyvtár. https://github.com/ageitgey/face_recognition

[18] Taigman, Y., Yang, M., Ranzato, M.A. and Wolf, L., 2014. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1701-1708).

[19] Schroff, F., Kalenichenko, D. and Philbin, J., 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 815-823).

[20] Deng, J., Guo, J., Xue, N. and Zafeiriou, S., 2019. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 4690-4699).

[21] He, K., Zhang, X., Ren, S. and Sun, J., 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

[22] Grother, P., Ngan, M. and Hanaoka, K., 2019. *Face Recognition Vendor Test (FVRT): Part 3, Demographic Effects*. National Institute of Standards and Technology.

[23] Labeled Faces in the Wild adathalmaz, http://vis-www.cs.umass.edu/lfw/

[24] FaceScrub adathalmaz, http://vintage.winklerbros.net/facescrub.html

[25] Deng, J., Guo, J., Ververas, E., Kotsia, I. and Zafeiriou, S., 2020. Retinaface: Single-shot multi-level face localisation in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 5203-5212).

[26] Using Histogram of Oriented Gradients (HOG) for Object Detection, https://iq.opengenus.org/object-detection-with-histogram-of-oriented-gradients-hog/

[27] , K., Kamaruddin, M.K., Nasir, H., Safie, S.I. and Bakti, Z.A.K., 2014, August. A comparative study between LBP and Haar-like features for Face Detection using OpenCV. In *2014 4th International Conference on Engineering Technology and Technopreneuship (ICE2T)* (pp. 335-339). IEEE.

[28] Wikipedia, *Fingerprint*, https://en.wikipedia.org/wiki/Fingerprint

[29] Wikipedia, Alphonse Bertillon, https://en.wikipedia.org/wiki/Alphonse_Bertillon

[30] BBC, *The teenage radio enthusiasts who helped win World War II*, 2013, https://www.bbc.com/news/technology-23162846

[31] *GDPR Adatvédelem mindenkinek - Adatvédelmi bírságok Magyarországon 2020-ban*, 2020, https://gdpr.blog.hu/2020/06/05/adatvedelmi_birsagok_magyarorszagon_2020-ban

[32] Yi, D., Lei, Z., Liao, S. and Li, S.Z., 2014. Learning face representation from scratch. *arXiv preprint arXiv:1411.7923*.

[33] VGGFace adathalmaz, https://www.robots.ox.ac.uk/~vgg/data/vgg_face/

[34] VGGFace2 adathalmaz, https://github.com/ox-vgg/vgg_face2

[35] Guo, Y., Zhang, L., Hu, Y., He, X. and Gao, J., 2016, October. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In European conference on computer vision (pp. 87-102). Springer, Cham.

[36] King, D.E., 2015. Max-margin object detection. arXiv preprint arXiv:1502.00046.

[37] Wikipedia, *Nyilvános kulcsú rejtjelezés*, https://hu.wikipedia.org/wiki/Nyilv%C3%A1nos_kulcs%C3%BA_rejtjelez%C3%A9s

[38] Wikipedia, *Homomorphic encryption*, https://en.wikipedia.org/wiki/Homomorphic_encryption

[39] Wikipedia, *Paillier cryptosystem*, https://en.wikipedia.org/wiki/Paillier_cryptosystem

[40] Nalini K. Ratha, Jonathan H. Connell, and Ruud M. Bolle, *An Analysis of Minutiae Matching Strength*, IBM Thomas J. Watson Research Center

[41] Eduardo Lopes Cominetti, Marcos Antonio Simplicio Junior, *Fast Additive Partially Homomorphic Encryption from the Approximate Common Divisor Problem*, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY 2020

[42] Wikipedia, Shor's algorithm, https://en.wikipedia.org/wiki/Shor%27s_algorithm

[43] Wikipedia, Quantum computing, https://en.wikipedia.org/wiki/Quantum_computing

[44] Jesse Damiani, *A Voice Deepfake Was Used To Scam A CEO Out Of $243,000*, 2019, Forbes, https://www.forbes.com/sites/jessedamiani/2019/09/03/a-voice-deepfake-was-used-to-scam-a-ceo-out-of-243000

[45] GDPR. 2018 reform of EU data protection rules. European Comission. May 25 2018.

[46] Földes Ádám, *Képfelvétel készítése tüntetésen*, Társaság a Szabadságjogokért, https://tasz.hu/files/tasz/imce/kepfelvetel_tuntetesen.pdf

Kiséry Máté Soma a Budapesti Műszaki és Gazdaságtudományi Egyetemen szerzett BSc villamosmérnöki diplomát 2020 januárjában. Jelenleg a villamosmérnöki MSc diplomamunkáján dolgozik, amelynek témája az arcfelismerés adatvédelmi kérdéseinek a vizsgálata. Eddigi tanulmányai során foglalkozott különféle táblázatos adatok anonimizálásával, valamint arcfelismeréshez kapcsolódó alkalmazási területek vizsgálatával. Emellett releváns tapasztalatot szerzett autóipari biztonságkritikus beágyazott rendszerekhez kapcsolódó szoftverfejlesztés területén.

Fábián István 2018-ban szerzett gépészmérnöki MSc diplomát a Budapesti Műszaki és Gazdaságtudományi Egyetemen, majd 2020 óta a Villamosmérnöki és Informatikai Karon az Automatizálási és Alkalmazott Informatikai Tanszék doktorandusza. Kutatási területe a gépi tanulás és az azzal kapcsolatos adatvédelmi kérdések, azon belül jelentős fókusszal az arcfelismerés és biometrikus azonosítás.

Gulyás Gábor György 2015-ben szerzett PhD fokozatott a Budapesti Műszaki és Gazdaságtudományi Egyetem Hálózati Rendszerek és Szolgáltatások Tanszékén. 2015 és 2018 között poszdoktori kutató és kutatómérnök volt a Privatics csapatban az INRIA-nál (Franciaország); 2019 óta tudományos munkatárs a BME Automatizálási és Alkalmazott Informatikai Tanszéken. Az adatvédelem informatikai kérdéseivel 2005 óta foglalkozik aktívan. Főbb szakmai érdeklődési és kutatási területei: anonimizálás és de-anonimizálás, webes megfigyelés adatvédelmi kérdései, gépi tanulás adatvédelmi kihívásai és a GDPR szabályozás.

Függelék

Technológia	Működés alapja	Előnyök	Hátrányok
Arcfelismerés	Az arc egyedi jellegzetességei.	+ Nincs szükség fizikai kontaktusra. + Könnyen megvalósítható, az arclenyomatok könnyen és gyorsan ellenőrizhetőek.	- A környezeti változásokra érzékeny lehet (megváltozott megvilágítás, más szögből készített képek stb.) - Külső jegyek megváltozása befolyásolhatja az eredményt (életkor, hajhossz, hajszín stb.)
Ujjlenyomat ellenőrzés	Az ujjlenyomat mintázata.	+ Gyors, megbízható, széles körben ismert. + Nem függ a környezeti viszonyoktól.	- Az ujjlenyomat az idő során degradálódhat (hosszú időn át végzett kétkezi munka, égési sérülések, vágások stb.) - Fizikai kontaktust igényel.
Írisz felismerés	Az írisz mintázata.	+ Nincs szükség fizikai érintkezésre. + Az írisz mintázata nem változik az öregedés során.	- Speciális infravörös kamera szükséges a működéséhez. - A kamerához nagyon közeli kontaktusra van szükség.
Tenyér véna mintázat felismerés	A tenyér vénarendszer mintázata.	+ Az ujjlenyomat olvasással szemben nem igényel fizikai kontaktust. + A pontosságát nem befolyásolják piszkok, vágások, sérülések, nedvesség a tenyéren.	- A vénamintázat az életkor előrehaladtával változhat. - Külső környezeti hatások (pl. megvilágítás) befolyásolhatják a pontosságot.

Technológia	Működés alapja	Előnyök	Hátrányok
DNS	A DNS szerkezetének vizsgálata.	+ Nagyon magas pontosság érhető el vele. + Viszonylag könnyű a mintavételezés (mintavételi pálca), bár kontaktust igényel.	- Drágább egy-egy mintát kiértékelni, mint más biometrikus rendszerek esetén. - Nehezebben skálázható (pl. arcfelismerést tömegeken is lehet alkalmazni).
Testtartás felismerés	Testtartás, járásmód egyedisége.	+ Kontaktus nélküli, távoli azonosítási mód. + Kevésbé érzékeny a képminőségre, mint más technológiák (pl. arcfelismerés).	- Külső (ruházat, cipő stb.) illetve belső (betegségek, sérülések stb.) faktorok befolyásolhatják a pontosságát.
Hang felismerés	Hang, beszédstílus egyedisége.	+ Nem igényel fizikai kontaktust. + Kényelmes, olcsó eljárás.	- Külső hatások (pl. betegség) megváltoztathatják emberek hangját. - Bizonyos környezetekben nem használható (pl. hangzavarban).
Aláírás felismerés	Aláírás dinamikájának felismerése (írási sebesség, toll nyomás stb. érzékelése)	+ Nehéz meghamisítani, mivel nem csak az aláírás külalakját, hanem keletkezésének módját (pl. toll nyomás, időzítés) is figyelembe veszi a rendszer.	- Külső hatások (pl. sérülések), illetve öregedés is befolyásolhatják a pontosságát.

Táblázat 2 – Néhány biometrikus azonosítási eljárás előnyeinek és hátrányainak bemutatása.