OKOSVÁROS-TECHNOLÓGIÁK III.

Az okos város kameraképeinek elemzése

SZŰCS GÁBOR
SALLAI GYULA

dialóg Campus

Szűcs Gábor – Sallai Gyula

Az okos város kamerakképeinek elemzése

Okosváros-technológiák
A technológia fejlődésének irányai és hatása
III. kötet

Sorozatszerkesztő
Sallai Gyula

Szűcs Gábor – Sallai Gyula

AZ OKOS VÁROS KAMERAKÉPEINEK ELEMZÉSE

Dialóg Campus Kiadó ❖ Budapest

A mű a KÖFOP-2.1.2-VEKOP-15-2016-00001 azonosítószámú, „A jó kormányzást megalapozó közszolgálat-fejlesztés” elnevezésű kiemelt projekt keretében működtetett 2017/162/BME-VIK számú, Okos város – okos közigazgatás elnevezésű Államtudományi Kutatóműhely keretében, a Nemzeti Közszolgálati Egyetem felkérésére készült.

Szakmai lektor

Magyar Gábor

© Dialóg Campus Kiadó, 2019
© Szerzők, 2019

A mű szerzői jogilag védett. Minden jog, így különösen a sokszorosítás, terjesztés és fordítás joga fenntartva. A mű a kiadó írásbeli hozzájárulása nélkül részeiben sem reprodukálható, elektronikus rendszerek felhasználásával nem dolgozható fel, azokban nem tárolható, azokkal nem sokszorosítható és nem terjeszthető.

Tartalom

Bevezetés 7

1. Mozgóképelemzés kameráknál 11

1.1. A kamera relatív viszonyának azonosítása 11

1.2. A kameramozgás felismerése 14

1.3. Perspektívakorrekció 18

1.4. Kamera által vett mozgó objektumok nyomkövetése 19

2. Gépi tanulás a multimédia-elemzéshez 23

2.1. A gépi tanulás típusai 23

2.2. A rejtett Markov-modell tanítása 25

2.3. Párhuzamos architektúrák a gépi tanulásban 28

3. A kamera képeinek feldolgozása 31

3.1. Morfológiai képfeldolgozás 31

3.2. Képi hisztogram 32

3.3. Éldetektálás képeken 33

3.4. SIFT-jellemzők a képeken 35

4. Videótartalmak rendszerezése 39

4.1. Csoportosítási szempontok 39

4.2. Multimodalitás kérdése 41

4.3. Metaadat-alapú megközelítés 42

4.4. Tartalomalapú megközelítés 43

4.4.1. Képitartalom-alapú megközelítés 44

4.4.2. Hangalapú megközelítés 44

4.4.3. Kombinált megközelítés 45

5. Arcfelismerés kameraképeken 47
5.1. Arcfelismerés alapfogalmai 47
5.2. Arcdetektálás 49
5.3. Érzelemfelismerés arcokon 50
6. Videótartalmak elemzése, áttekintése 53
6.1. Videószegmentálás 54
6.2. Összegző jelenetsor készítése 56
6.2.1. Videókivonatoló módszerek 56
6.2.2. Kiválasztás felügyelet nélküli gépi tanulással (klaszterezéssel) 57
6.2.3. Kiválasztás felügyelt gépi tanulással (osztályozással) 59
6.2.4. Végső összefűzés 60
7. Keresés videótartalmakban 63
7.1. Tartalomalapú képvisszakeresés 63
7.2. Képalapú keresés videótartalmakban 64
7.2.1. Invertált index 65
7.2.2. A képinformációk tárolása 67
8. Kihívások és trendek a multimédiában 69
8.1. Az okos város kameraképeinek feldolgozása 69
8.2. Az okos várossal kapcsolatos médiatechnológiák 71
8.3. Az okos várost érintő közösségi média kutatása 73
Irodalom 75

Bevezetés

Az okos várossá fejlesztés, az okosváros-koncepció megvalósítása során egyszerre figyelünk többek között a lakosság életminőségének javítására, a város hatékony irányítására, a közbiztonság fejlesztésére, a zöld és megújuló energiák hasznosítására. E célok megvalósításában a fejlett digitális technológiák alapvető szerepet játszanak, meghatározó módon befolyásolják a kiaknázható lehetőségeket. A városirányításnál az okos város digitális megoldásai fókuszálhatnak a tömegközlekedés valós idejű optimalizálására (például a légszennyezettség minimalizálása érdekében), a város üzemeltetésével kapcsolatos feladatokra, illetve a vészhelyzeteknél a beavatkozó szervek minél gyorsabb és hatékonyabb riasztására. Az ilyen megközelítésű megoldások átfogóan szemlélik a városi élet valamennyi szegmensét, kapcsolódjon az akár a városi lakossághoz (egyéni vagy közösségi szinten) vagy szervezetekhez (önkormányzati, vállalati, civil funkciókkal). Az okos városnál különösen fontosak az olyan életminőséget közvetlenül érintő *smart* alkalmazások, mint például okos egészségügyi alkalmazások, idősgondozás/-felügyelet, okos oktatási, kulturális és személyes szolgáltatások, okos környezetvédelmi, köztéri és közbiztonsági megoldások, amelyek a digitális technológia révén intelligensebbek, nagyobb hozzáadott értéket képesek nyújtani, így érve el, hogy az „okos várossá válás" egyben magasabb életminőséget jelentsen. Jó néhány ilyen fejlesztésnél felmerül az igény képek, mozgóképek, multimédia-tartalmak intelligens felhasználására. Gondoljunk csak a térfigyelő kamerákra a közbiztonságnál, arc- és alakfelismerésre egy idősgondozási távfelügyeletnél, a sofőr monitorozására a közlekedésnél, multimédia-tartalmak intelligens feldolgozására a szórakozásnál, digitális audiovizuális technológiákra az okos város épületeinél. Jelen kismonográfia célja az effajta technológiák és felhasználási lehetőségeik bemutatása.

Egy épületbe számos ponton beépíthetők a legmodernebb digitális audiovizuális technológiák (kamerák, kijelzők, kivetítők, hangfalak vagy az ezeket összekötő rendszerek stb.), amelyek lakóépületek esetén a szórakozást, munkahelyi irodaépületek esetén a munka hatékonyságának javítását, oktatási intézmények esetén pedig a tanítást szolgálják. Ez utóbbi esetben a hagyományos értelemben vett táblák helyett digitális eszközökre lehet

alapozni az oktatási programot, amely nemcsak az aktuálisan elhangzó előadás színvonalához járul hozzá, hanem könnyen rögzíthetővé, visszajátszhatóvá, megoszthatóvá is teszi az elhangzottakat – gyakorlatilag létrehozva egy további költségek nélkül tárolható audiovizuális tudástárat. A tanulási módszerek is megváltoztak manapság: a videóanyagok rengeteg információt tudnak akár rövid időbe is sűríteni. A mai generációk az okostelefonokkal gyakorlatilag egy miniatűr stúdiót hordanak a zsebükben, az információgenerálás és -megosztás eszközét és lehetőségét. Egy digitális könyvtár létrehozásához, szemben a szokásos akár több ezer négyzetméteres helyigényű hagyományos könyvtárral, jóval kisebb terület szükséges, ez pedig konkrét megtakarítást jelent már a beruházás szintjén is.

Ilyen rendszerek kiépítése meghatározza az épület kialakítását, átgondolása nem várhat a belsőépítészeti munkafázisig. A digitális tudástár létrehozása és kezelése például szükségessé teszi egy stúdióhelyiség építését, a különböző helyiségek, termek összekapcsolását. Az audiovizuális, interaktív technológiák a sportlétesítményekben is egyre inkább teret nyernek, habár a legtöbbször még hiányzik a tartalomgenerálás és -fogyasztás emelt szintje, a felhasználók individuális kiszolgálása. A nézők szórakoztatása mellett sportszakmai támogatórendszerek is működnek. Ennek a mérkőzések, edzések rögzítése, visszanézhetősége mellett további előnyei is lehetnek. A mérkőzés során egyszerre több kamera veszi a játékteret, s teszi elemezhetővé a labda és a játékosok mozgását is. És nemcsak épületen belül, hanem közösségi terek, több épületet átfogó városrészek vonatkozásában hozhatjuk a példákat, az igényeket erre a technológiára. [Mizsei 2016]

E tanulmányban az okos város kameraképeinek elemzésével foglalkozunk. A tanulmányban a filmes szaknyelv kifejezéseit használjuk. Az eredeti angol szavak fonetikus átirata már régen beépült a magyar nyelvbe, szakmai közegben kizárólag ezek élnek. Ugyanakkor egyes fogalmak idegen nyelvű és a magyar jelentésének eltérésére is szeretnénk felhívni a figyelmet, ami esetünkben különösen fontos. A kamera szó írásmódjában már régen „magyarosodott". Sok más nyelven, így a műszaki szaknyelv szempontjából kitüntetetten fontos angol nyelven is a *camera* fényképezőgépet *(camera obscura)* és mozgóképfelvevő készüléket is egyaránt jelent. A mai magyar nyelvben azonban a kamerát csak filmfelvevő gép és videokamera értelemben használjuk, így használjuk kismonográfiánkban is. Az állókép készítésére alkalmas eszköz megnevezése nyelvünkben: fényképezőgép.

Tanulmányunk alapfogalma a kamera, a szűkebb, magyar értelemben. Sokféle kamera létezik. A mai kamerák felbontása már minimum HD-kategóriába

esik, azaz 1280×720 pixel, illetve full HD esetén 1920×1080 pixel felbontásúak; egy kisebb részük ennél is nagyobb felbontással (4K felbontással, amely 3840x2160 pixelt jelent) rendelkezik, a professzionális szintet, a csúcstechnológiát a 8K felbontás (7680×4320 pixel) jelenti. A felbontás mellett a kamerák másik fontos paramétere a képfrissítési sebesség, amelyet képkocka/másodpercben szoktak megadni *(fps, frame per second)*. Tipikus fps-paraméterértékek a kamerák esetében: 25, 50, 60; a sportkamerák ennél magasabb értékekkel rendelkeznek: 120, sőt 240 fps is előfordul; azt viszont figyelembe kell venni, hogy nagyobb fps-érték esetén a felbontás általában csökken, ezért nem lehet olyan szép felvételt készíteni, mint a normál felbontás esetén. Egy magas paraméterértékű kamera használata esetén azonban lehetőségünk van lassított felvételek rögzítésére is elmosódás nélkül. Érdemes megemlíteni a bitsebesség-paramétert is, az átlagos bitsebesség 10 és 100 Mbit/s között mozog. A kamerák felvételeinél meghatározó körülmény a megvilágítás szükséges mértéke is. Egy átlagos kameránál a megvilágításnál 0,5 lux is elegendő, azonban egy térfigyelő kamera esetén ez a paraméter jóval alacsonyabb, lehet akár 0,01 lux is, és az infraredes kamerák képesek a teljes sötétben is jól látható felvételeket készíteni.

Tanulmányunkban a kamera által felvett tartalmak elemzésének technikáit vizsgáljuk. Természetesen a vizsgálódást rengeteg szempontból megtehetnénk (például mélységélesség, telítettség, képegyensúly, világítás kontrasztossága, világítás összhatása), így a teljesség igénye nélkül kiválasztottuk az okosváros-alkalmazások körében általunk legfontosabbnak tartottakat. A kamerák helyzete szempontjából külön figyelmet szentelünk a kameramozgás felismerésére és a kamera által vett mozgó objektumok nyomkövetésére. A kamerákból nyert multimédia-tartalmak elemzéséhez fontos segítséget nyújtanak a mesterséges intelligencia módszerei, azon belül is a gépi tanulás, amelynek külön fejezetet szánunk. A képfeldolgozás témakörben bemutatjuk a morfológiai képfeldolgozást, képi hisztogramokat, az éldetektálást és egyéb képi jellemzőket, amelyek a képből kinyerhetők. A képeken kívül foglalkozunk a mozgóképekkel is, amelyekre a videó szót szinonimaként használjuk. Áttekintjük a videótartalmak rendszerezésének megoldásait, tisztázzuk a multimodalitás fogalmát. Bemutatjuk a mozgóképelemzés két alapvető módját, a metaadat-alapú és tartalomalapú megoldásokat. Tartalom szempontjából a felhasználók tudni szeretnék, hogy ki vagy mi szerepel a képen/videón. A látható személyek arc alapján felismerhetők gépi úton is, ezért az arcfelismeréssel részletesen foglalkozunk. Az okos város polgárai által készített *homemade* videók rendszerezéséhez, gyors

áttekintéséhez rendelkezésre álló számítógépes lehetőségek közül bemutatjuk a videószegmentálást, az összegző jelenetsor készítését, videótartalmakban való keresési eszközöket, a tartalomalapú képvisszakeresést. Tanulmányunkat a multimédia területén felmerülő nehézségek és kihívások áttekintésével, a jelenlegi legégetőbb kérdések és a várható fejlődési irányok ismertetésével zárjuk, amelyeket elsősorban az okosváros-alkalmazások oldaláról világítunk meg. A tanulmány végén a felhasznált irodalmak jegyzéke található.

A szerzők[*]

[*] Dr. Szűcs Gábor, okleveles villamosmérnök, informatikai PhD, a Budapesti Műszaki és Gazdaságtudományi Egyetem egyetemi docense. Szakmai és tudományos életpályáját szimuláció és mesterséges intelligencia kutatásával kezdte; jelenlegi kutatási területe a médiainformatika, valamint a mesterséges intelligenciához tartozó gépi látás és mély tanulás.
Prof. Dr. Sallai Gyula okleveles villamosmérnök, a Magyar Tudományos Akadémia doktora, a Budapesti Műszaki és Gazdaságtudományi Egyetem professor emeritusa. Szakmai és tudományos életpályája során a távközlés, illetve az infokommunikáció különböző szakterületein dolgozott. Az utóbbi években fő érdeklődési területe a jövő-internet-technológiák és azok okosváros-alkalmazásai.

1. Mozgóképelemzés kameráknál

Az okos város kameraképeinek elemzéséhez rengeteg szempontot, aspektust figyelembe vehetünk: a kép vizsgálatával megállapítható a mélységélesség, a telítettség, a képegyensúly, a kameramozgás (ha mozgó kameráról van szó), mozgó tárgy követése, a világítás kontrasztossága, a világítás összhatása stb. A kamerák lehetnek fixen rögzítettek, mint a térfigyelő kamerák esetében, de lehetnek mobilisek is. Ez utóbbira példa a polgárőrség vagy az útfenntartó járművére szerelt mozgó kamera. Nemcsak városi közcélú kamerákból jöhet tartalom, hanem a városi polgárok által készített kézi kamerákból, mobiltelefonokba beépített kamerákból is, így nemcsak a kép, hanem a hang is elemezhető. Hang esetén beszélhetünk az éppen hallható hangról vagy zörejről, narráció jelenlétéről, zenéről, illetve a képet kísérő zenei aláfestésről stb. A multimédiás tartalmak száma napjainkban nagymértékben növekszik. Ezek rendszerezése, keresése vagy elemzése jelenleg nem túl hatékony, nem minden esetben megoldott, és a fent említett szempontok még komplexebbé teszik az ezzel kapcsolatos feladatokat, problémákat, amelyek esetében ráadásul elvárás a különféle elemzési részfeladatok automatizálása.

1.1. A kamera relatív viszonyának azonosítása

A film-, a tévé- és videóiparban a mozgóképnek azt a tulajdonságát, amely a szereplők és a környezetük egymáshoz való viszonyát, elhelyezkedését, illetve mozgását fejezi ki, plánnak nevezzük. A pontos terminológia a gyártási közeg függvényében változhat, de az alapvető jelentések megegyeznek: [MediaCollege]

- Nagytotál plán: a hangsúly a környezet bemutatásán van, a szereplők, ha láthatók egyáltalán, akkor 30 méternél messzebb látszanak.
- Totál plán: a szereplők már jobban láthatók, de a hangsúly még mindig a környezetükön van.
- Kistotál plán: a szereplők vagy egy nagyobb tárgy teljes egészében láthatók, kitöltik a képet, amennyire lehetséges.

- Amerikai plán: a szekond plán egy variációja, a szereplőket térdtől felfelé mutatja. Más néven cowboyplán, a westernekből eredő elnevezés lényege, hogy a szereplő oldalán függő fegyver benne legyen a képben.
- Szekond plán: egy vagy több szereplő fejét és testének egy részét, illetve valamilyen közepes méretű dolgot mutat meg.
- Premier plán: egy egész fejet, testrészt vagy valamilyen kisebb méretű dolgot jelenít meg.
- Szuper közeli plán: a premier plánnál még közelebbi állás, a szereplő fejének csak egy részlete látszik.

Filmes és videós környezetben többféle kameramozgás lehetséges, alább felsoroljuk az alapvető mozgásokat.

- Statikus: a kamera nem mozog.
- Svenk *(pan):* a kamera vízszintes balra-jobbra forgatása.
- Fahrt *(dolly és truck):* a kamera mozgatása vízszintes síkban. Mozgó karakter, tárgy követése, általában párhuzamosan, illetve a kamera oldalirányú mozgás nélküli közelítése, távolítása, miközben a fókusztávolság nem változik.
- Svenkfahrt: kombinált kameramozgás, svenk és fahrt együttes alkalmazása.
- Billentés *(tilt):* a kamera vízszintes síktól eltérő, felfelé-lefelé döntése.
- Daruzás *(pedestal):* a kamera vízszintes síkban, helyben marad, csak függőlegesen emelkedik vagy süllyed.

Egy kép dinamikáját a képen látható tárgyak, objektumok mozgásából állapíthatjuk meg, és ez lehet statikus vagy dinamikus. Egy kép statikus, ha az objektumok semmilyen mozgást nem végeznek, azonban kameramozgás megengedett. Ezzel szemben egy kép dinamikus, ha az objektumok mozgásban vannak.

Az egyes aspektusok felismerésére olyan jellemzőket kell előállítani, amelyek a vizsgált aspektust jól jellemzik, és megfelelően elkülöníthetők a különböző aspektusok egymástól. A kameramozgás és képdinamika felismerésére például hasonló jellemzővektor-készlet alkalmazható, mivel mindkét vizsgált tulajdonság a képen látható mozgásokat jellemzi. A plánok felismerésére külön módszereket érdemes alkalmazni.

A kameramozgás és képdinamika esetében fontos segédmódszer az úgynevezett optikai áramlás, amely a háromdimenziós tér kétdimenziós képtérbe vetített felvételéből meghatározható elmozdulásmintázat. Ez lényegében egy

sebességvektorokat tartalmazó vektormező, amelyet a mozgókép egymás utáni képrészleteinek összerendeléséből kapunk meg. Az optikai áramlás a tárgyak, felületek, szélek látszólagos mozgásáról szolgáltat információt. A megfigyelő és a tárgyak relatív mozgásából következően ez a vizuális elmozdulás nem feltétlenül egyezik meg a valós elmozdulással. Egy optikai áramlás sűrű, ha a vektormező a kép minden pixelében megadja az elmozdulás nagyságát és irányát. Ritka optikai áramlást úgy kaphatunk, ha a képnek nem az összes, hanem csak néhány meghatározott pontját követjük.

Az optikai áramlás kiszámításához feltételezzük, hogy ha egy tárgy a kamerához viszonyítva elmozdul, akkor azonos intenzitás tartozik az objektum azonos pontjához, azaz a pontnak csak a pozíciója változik, az intenzitása konstans. Ezt nevezzük fényességmegmaradási kényszernek *(brightness constancy constraint).* Az optikai áramlásra kidolgozott Farnebäck-algoritmus [Farnebäck 2003] például úgy dolgozik és azon alapul, hogy egy videószegmens szürkeárnyalatos képek egymásutánjából áll, amelyekben az egyes szürke struktúráknak adott orientációik vannak. Farnebäck egy mozgástenzort definiál úgy, hogy az a legkisebb maradékot használja fel az input (képsorozat) és egy meghatározott bázis között a legkisebb négyzetek módszere szerint. Ez a mozgástenzor a választott bázishalmaz használatával írja le a képi struktúrák mozgását. Meg kell említeni ennél az algoritmusnál, hogy nagymértékű elmozdulások becslésekor a tenzoralapú megközelítések nehézségekbe ütköznek, azonban egy durva-finom felbontással, például piramisépítéssel ezek a korlátozások leküzdhetők. Az OpenCV-ben megvalósított Farnebäck-algoritmus sűrű optikai áramlást ad meg [Farnebäck 2003], azaz a kép minden pontjára kiszámolja az elmozdulásvektorokat. A kameramozgás felismeréséhez azonban elegendő lehet, ha csak a képen látható fontosabb pontokat követjük. Ezért a vizsgált képkocka feldolgozását először a képen látható fontosabb pontok megtalálásával kell kezdeni, majd ezekre a pontokra lehet optikai áramlást számítani. A fontos pontok megtalálására nagyon jól használhatók a SIFT- [Lowe 2004] vagy a SURF-leírók. [Bay et al. 2006]

A SURF *(Speeded-Up Robust Features)* [Bay et al. 2008] skála- és forgatásinvariáns-jellemző felismerő. Sokféleképpen felhasználható, például objektumfelismerésre, osztályozásra, 3D-rekonstrukcióra, objektumkövetésre, illetve képeken lényeges pontok kinyerésére. Részben a SIFT- *(Scale Invariant Feature Transform)* leíróra építkezik, de megismételhetőségben, pontosságban és robusztusságban is jobban teljesít. Az algoritmusnak két fő része van: az első a kulcspontkeresés, majd a kulcspont környezetének leírása. Ezután (harmadik részként) megemlíthető még a kulcspontok egymáshoz

illesztése is, de ez már nem tartozik szigorúan az algoritmushoz, illetve többféle módszerrel is elvégezhető a kulcspontok összehasonlítása a képeken.

A kameramozgások megállapításánál olyan módszereket (pontosabban elég hasonló módszereket) használnak, amilyeneket videószegmentálásra dolgoztak ki. Ezek közül több módszer az MPEG kódolt videó P és B képeiből kinyert mozgásvektorokat használja fel. [Nilesh–Ishwar 1997] [Lee–Hayes 2002] Egy másik módszer *block-matching* algoritmussal két vektort állít elő egy képre, amellyel a svenk- és zoommozgások felismerhetők. A kameramozgásokat egy *Hidden Markov Model* (HMM) állapotaiként értelmezi, a HMM ezután videószegmentálásra is használható. [Boreczky–Wilcox 1998]

A multimédiás tartalmak ilyenfajta elemzésére elkészült egy rendszer [Marosvári 2016], amely annotálni képes a videóban található kameramozgásokat. A rendszer egy multimédia-bányászati rendszer a különböző felismerendő tulajdonságokhoz egy-egy tanulóalgoritmussal. A tanulóhalmazt egy teljes filmből kinyert képkockák alkották, amelyekre többfajta jellemzőkinyerést hajtottak végre. A jellemzőkinyeréshez több képfeldolgozási algoritmust alkalmaztak: SURF-pontokat, optikai áramlást. A rendszer tovább bővíthető, például jelenetdetektálás vagy arcdetektálás részekkel, amelyekhez több programot, programkönyvtárat szükséges használni: ffmpeg-et [Ffmpeg] a rövid tesztvideók vágásához, illetve a jelenetdetekciókhoz, OpenCV-t a jellemzőkinyerő alkalmazáshoz, statisztikai eszközt az elkészült adatsorok és manuális annotációk összeillesztésére, illetve RapidMinert az osztályozáshoz (például neurális háló) és kiértékeléshez.

1.2. A kameramozgás felismerése

A kameramozgás felismerését el lehet végezni például egyetlen vektor kinyerésével (SURF-mozgásvektorral). Ez a mozgás főirányát jól jellemzi, azonban az egyirányú, de különböző kameramozgások elkülönítését nem teszi lehetővé (az oldalirányú svenk és fahrt, valamint a függőleges síkon történő billentés és daruzás elkülönítése ezzel nem lehetséges). Ezen kívül előfordulhat az is, hogy a képen látható tárgyak, szereplők mozgása a számított vektort is bemozgatja, azaz a számított vektor nem a teljes képre jellemző mozgást mutatja. Ezért többféle mozgásvektort érdemes kinyerni, hogy az előbb említett esetekre is megfelelően működhessen a kameramozgás felismerése. Ugyanakkor a SURF-pontvektor a képdinamika elemzésekor is hatékony lehet.

A többféle mozgásvektor számításának alapja a képekre számolt sűrű optikai áramlás. Az optikai áramlást egy adott pontban a vizsgált képkocka (y, x) pontja és a következő képkocka (y, x) pontja határozza meg. Az optikai áramlásból számított vektorok különböző módon történő összevonása alapján többféle jellemzővektor számítható, amelyeket a következőkben részletezzük.

Mozgásvektor SURF-pontokban [Bay 2006]: a módszer lényege, hogy előállít minden képkockára egy vektort úgy, hogy az optikai áramlásértékeket csak a képen talált SURF-pontokban veszi figyelembe: ezt a vektort nevezhetjük SURF-mozgásvektornak. A mintavételezett képkockán SURF-leírókat keres, majd a sorban következő képkockán csak azokat a SURF-leírókat tartja meg a módszer, amelyet az előző képkockán is megtalált. Ezzel biztosítható, hogy valóban csak a fontosabb pontok mozgását vegye figyelembe a módszer, a téves vagy eltűnő pontokat ne kövesse több képkockán keresztül. A SURF-leírók illesztéséhez először a SURF-leírókat kell kiszámolni, majd a két képkocka SURF-leíróit összehasonlítja az euklideszi távolság alapján.

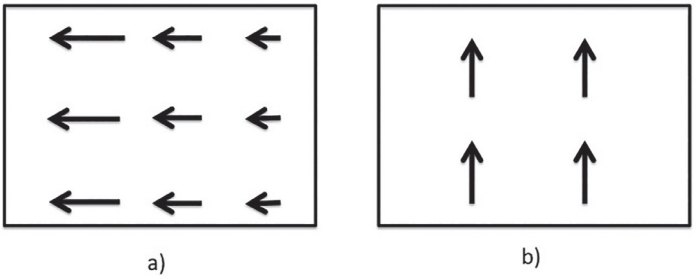
Hosszabb és mozgalmasabb jeleneteknél előfordul, hogy több képkocka után a követendő SURF-leírók illesztése már nehezebb, így a számuk képkockánként egyre csökken. Ennek kiküszöbölésére az a megoldás alkalmazható, hogy ha a SURF-leírók száma egy minimum alá csökken (vagy a jelenet első képkockáján talált SURF-leírók számának adott részaránya alá csökken), akkor újra az összes SURF-pontot érdemes figyelembe venni. Előfordulhat olyan képkocka, képkockasorozat is, amelyen egyetlen SURF-pont sem található. Ez leginkább teljesen homogén képek esetén történhet, például sötét átmeneteknél, vagy teljesen fókuszálatlan, homályos vágóképeknél is. Ekkor a képet 3×3 egyenlő téglalapra osztja a módszer, és mindegyik téglalap középpontjában számított optikai áramlás alapján számítódik a vektor.

A vizsgált képkockákra az optikai áramlásból kapott vektorokat a megtalált és megtartott pontokban átlagolhatjuk, ezt az elmozdulásvektort pedig a videó képméretéhez normalizáljuk így megkapva a végső elmozdulásvektort, a SURF fő mozgásvektort. A vektornak a hosszát, illetve a vízszintes tengellyel bezárt szögét kell figyelembe venni. Ez a vektor többnyire jól jellemzi a fő mozgás irányát és nagyságát, megállapítható belőle, hogy van-e kameramozgás, vagy a mozgás statikus-e; illetve a kameramozgás irányát is jól jellemzi.

A képkocka kilenc részre osztásával egy finomabb módszer alakítható ki. [Marosvári 2016] A további jellemzővektorokat úgy határozhatjuk meg, hogy a vizsgált képet felosztjuk 3×3 egyenlő téglalapra, és a téglalapban található pixelekre számolt mozgásvektorokat átlagoljuk. Így tehát kilenc

vektort kapunk egy képhez, amelyek közvetlenül felhasználhatók a különböző kameramozgások elkülönítésére (svenk vagy fahrt, billentés vagy daru). Továbbá ezekből a vektorokból még egy teljes képre jellemző vektor is előállítható: a kilenc vektor mediánjának számításával (mediánvektor). Ez a vektor a SURF fő mozgásvektorral azonos funkciót tölt be, azaz a főbb mozgást hivatott detektálni, azonban azzal ellentétben a képen látható egyes kisebb részek erős mozgása kevésbé hat erre a vektorra, így pontosabban mérhető vele az általános mozgás (átlagszámítás helyett a medián használatánál egy-egy mozgalmasabb képrészlet kevésbé hat a számított vektorra).

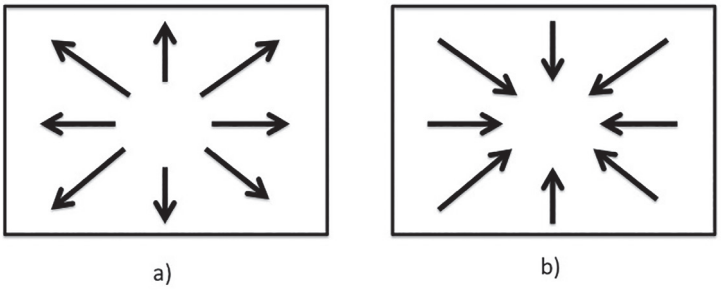
A kilenc vektorral az egyes kameramozgások a következő módszerrel különíthetők el. A detektált mozgásokat a dimenziók mentén három különböző csoportra oszthatjuk: oldalirányú mozgás (svenk vagy fahrt) (lásd 1.a ábra), függőleges irányú mozgás (billentés vagy daru) (lásd 1.b ábra), illetve előre-hátra történő mozgás (előre- vagy hátrafahrt). Az oldalirányú és függőleges mozgás jelenléte már a SURF fő mozgásvektorból vagy a mediánvektorból detektálható, az irányt egyértelműen kijelölik ezek a vektorok. Azonban előre- vagy hátrafahrt detektálásához meg kell vizsgálni külön a kilenc vektor egymáshoz képesti helyzetét. Előrefahrt esetén a külső vektorok a kép középpontjától kifele mutatnak, hátrafahrt esetén pedig a kép középpontjába mutatnak, ahogy ez a 2. ábrán látható.



1. ábra

Oldalirányú és függőleges irányú kameramozgás

Forrás: a szerzők szerkesztése



2. ábra

Előre- és hátrafahrt esetén látható vektorok a kameramozgásnál

Forrás: a szerzők szerkesztése

E mintázat felismerésével ezek a mozgásfajták detektálhatók. Azonban a gyakorlatban ritkán jelennek meg ilyen szépen, szabályszerűen ezek a vektorok, ezért a mintázatot nem mind a nyolc külső vektor figyelembevételével érdemes keresni, hanem érdemes a vektorokat páronként összehasonlítani. A négy sarki vektor közül mindig két azonos oldalon levőt kell összehasonlítani, azaz a vizsgált vektorpárosítások: 1–7, 1–3, 3–9, 7–9. Amennyiben bármelyik párosnál megfigyelhető az adott mozgásra utaló minta (kifelé vagy befelé mutatnak, az iránynak megfelelő 90°-os szögtartományban), akkor hátra- vagy előrefahrtnak detektálhatjuk a mozgást.

E módszernek azonban van hátránya is. Az oldalak közötti sebességkülönbségek csak viszonylag közeli tárgyak felvételénél észlelhetők, egy nagytotál esetén ritkán. Valamint svenkelés (és billentés) esetén, ha a kamera teljesen szemben vagy majdnem szemben áll, akkor az oldalak sebessége nagyjából megegyezik, tehát a módszer fahrtnak (daruzásnak) fogja felismerni. Ezt kiküszöbölni a kameramozgások utólagos elemzésével lehetne, ahol a svenk-fahrt-svenk időbeli felismerési mintázatból detektálni lehetne egy valószínűleg téves fahrtfelismerést.

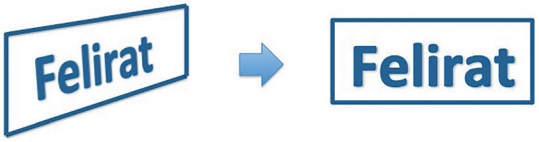
A kinyert fő vektorok (SURF fő mozgásvektor, mediánvektor) a kameramozgás felismerésén túl a képdinamika meghatározására is felhasználhatók: bináris esetben ehhez tulajdonképpen elegendő egy alacsonyabb küszöbértéket megállapítani a vektorok nagyságára; nem bináris esetben pedig a fő vektor hossza adja meg a dinamikai attribútumot.

1.3. Perspektívakorrekció

Gyakran előfordul, hogy az objektum, amit a kamerán keresztül vizsgálunk nem tökéletesen merőleges a kamerára, hanem bizonyos szögből látszik. Ilyenkor lép fel a perspektíva torzulása, amivel gyakran találkozunk, de képfeldolgozás szempontjából nem kívánatos jelenség. Ennek megoldására jöttek létre a perspektívát korrigáló algoritmusok [Mohan et al. 2008] [Jagannathan–Jawahar 2005], amelyek olyan korrigált képeket állítanak elő, amelyeket már könnyebben fel lehet dolgozni.

Egy síkfelület perspektívatorzulását úgy lehet felfogni, mint egy projektív transzformációt a síkfelületen. A projektív transzformáció egy általánosított lineáris transzformáció (homográfia) a homogén koordinátarendszerben definiálva, amely három komponens szorzatából áll: hasonlóságkomponens, affin komponens és egy projektív komponens szorzatából. A célunk, hogy a projektív és az affin komponenseket eltüntetve csak a hasonlóság komponense maradjon. Ehhez a korrigáláshoz a képen sok minden segítségül szolgálhat: a dokumentum széle, a szöveg dőlése, a lapkiosztás stb. Párhuzamosok és merőlegesek detektálásával az algoritmus a következő lépésekkel írható le:

- Kell találni egy pár párhuzamos egyenest, amelyek metszik egymást a perspektíva torzított képen.
- A pontokon végighaladva fel kell állítani a megtalált két egyenes egyenletét, és helyre kell állítani a képet a projektív komponenst eltávolítva.
- Azonosítani kell a képen két merőleges egyenest és el kell távolítani az affin komponenseket.



3. ábra

Perspektívakorrekció

Forrás: a szerzők szerkesztése

Ha egy felszerelt kamera esetén szeretnénk ezt a perspektívakorrekciót megvalósítani, akkor az első lépés a kamera betekintési szögéből adódó perspektíva korrigálása. A korrekció után a kép majd úgy fog látszani,

hogy a téglalapok oldalvonalai párhuzamosak lesznek, ahogy az a 3. ábra jobb oldalán látszik a korrekció után. Ez a fajta művelet négy kontrollpontot használ a transzformáció elvégzéséhez. Manuális esetben ezt a négy pontot a felhasználó állítja be a képi beállításoknál.

1.4. Kamera által vett mozgó objektumok nyomkövetése

A kamera által vett mozgó objektumok nyomkövetésénél kihasználhatjuk azt a tényt, hogy a közvetlenül egymást követő képkockák általában nagymértékben megegyeznek – ami nyomkövetésnél a számításokat nagyban megkönnyíti – így sok esetben elegendő csak a változásokat figyelemmel kísérni. Ezen a megfigyelésen alapul a későbbiekben bemutatandó metódus is, amely lehetővé teszi a számítógépes rendszerek számára, hogy megtalálják és kövessék a megfelelő objektumokat. Ezen felül az objektumok detektálása során meg kell küzdenünk azzal a problémával, hogy a kamerafelvételeken nagy valószínűséggel nem csupán a számunkra érdekes objektumok mozognak. Számolnunk kell a környezet változásaiból fakadó hibákkal, amelyeket okozhat a kamera mozgása (remegése) vagy a fényviszonyok megváltozása is.

Ahhoz, hogy a számunkra releváns objektumokat követni tudjuk, első lépésként detektálnunk kell azokat. Ez a művelet több komponensből épül fel, amelyeknek megválasztása nagyban befolyásolja a rendszerünk eredményességét, ugyanis a detektálás során határozzuk meg azokat a képtartományokat, más néven befoglaló dobozokat, amelyeket követnünk kell majd. A mozgó részek detektálásához lehet használni a háttérkivonást *(background subtraction),* amely egy általános eljárás arra, hogy képsorozatokon valós időben szegmentáljuk a mozgó területeket. Ez a metódus magában foglal egy referenciaképkiszámítást, majd a különbségi képek meghatározását, amelyeket úgy kapunk meg, hogy a referenciaképből kivonjuk az adott képkockákat és a különbségi képeken úgynevezett *thresholding* művelet eredményét láthatjuk.

A háttérkivonás eredménye egy olyan bináris szegmentálása a képnek/képkockának, ahol a mozgó objektumok ki vannak emelve. A referenciakép előállításának legegyszerűbb módja, ha a háttérképet időalapon átlagoljuk. Ennek hátránya, hogy ehhez az eljáráshoz szükséges egy tanulási periódus beiktatása, és előfordulhat, hogy a tanulás után a háttérobjektum mozogni kezd, miközben az előtérobjektum mozdulatlan marad. Másik megoldás egy

fix referenciakép készítése, amikor nincs egyetlen előtérelem sem a képen, csak a háttér látható.

Az objektumok kiemelését, detektálását követően az objektumokat folyamatosan követni szeretnék annak érdekében, hogy a későbbi azonosítási lépésben ugyanazt az egyedet feleslegesen ne azonosítsuk minden egyes képkockán. A járókelők, objektumok számlálásánál is fontos szerepe van a követésnek, ugyanis e nélkül lehetetlen lenne még csak közelítőleg is meghatározni a számukat. Az objektumok követésére alkalmas algoritmusok egyike a Kálmán-szűrő [Kalman 1960], amelyet széles körben alkalmaznak, egyszerű, kisebb járművek irányításától kezdve az űrhajók irányításáig.

A Kálmán-szűrő lényege, hogy megbecsüljük egy diszkrét idővezérelt folyamat állapotát, amely megadható egy lineáris sztochasztikus egyenlettel. Ez alapján a folyamat aktuális értéke leírható mint az előző pillanat értékének és egy kontrolljelnek a lineáris kombinációja, kiegészítve a feldolgozási zajjal. A Kálmán-szűrő ezen kívül magában foglal még egy egyenletet, amellyel minden mérési érték leírható úgy, mint a jel és a mérési zaj lineáris kombinációja.

A Kálmán-szűrős követés egy predikciós és egy korrekciós fázisból áll, s mindez kiegészül egy kezdeti inicializáló lépéssel. A Kálmán-szűrők inicializálása az első olyan képkockánál történik, ahol sikeres detektálás történt. Ehhez tehát egy objektumdetektáló szükséges, amelynek a kimenete a detektált objektumok középpontjai, illetve befoglaló dobozai. Minden egyes ilyen objektumhoz rendelhető egy Kálmán-szűrő-példány, így akár több külön objektum is nyomon követhető ennek segítségével. Az inicializálást követően predikciót végez a rendszer becslést adva arra, hogy az adott objektum a következő időpillanatban hova fog elmozdulni. Ezt a predikciót minden külön Kálmán-szűrő-példányra el kell végezni. A következő képkockán újból elvégezhető a detektálás, amelynek eredményei lesznek az úgynevezett „mérések”, és ezeket felhasználva a Kálmán-szűrőn elvégezhetjük a korrekciót. Ezt a kis ciklust szisztematikusan végezve folyamatosan nyomon követhetjük a kamerák által vett mozgó objektumok (legyenek azok járművek vagy személyek) pályáját.

Ahhoz, hogy a kamera helyzetén kívül komoly elemzési statisztikákat tudjunk készíteni, az egyik lehetséges módszer, ha közvetlenül a videót vizsgáljuk meg, vagy a szemantikus metaadatokból gyűjtjük ki a statisztikához szükséges információkat. Bizonyos esetekben a szemantikus metaadatok nehezen hozzáférhetők, illetve egyáltalán nem léteznek (például biztonsági térfigyelő kameráknál), ilyenkor a videóstream (videó-médiafolyam)

közvetlen vizsgálata az egyetlen járható út. Ennek a közvetlen vizsgálatnak az elvégzéséhez tudnunk kell, hogy milyen jellemzőket kell a videóból kinyerni a megfelelő felismeréshez, tisztában kell lenni a szemantikus elemzéshez szükséges jellemzőkkel.

A kamerákból és már forrásokból keletkező multimédia-tartalmak elemzéséhez (különösen a szemantikus elemzéshez) fontos segítség lehet a gépi tanulás. A gépi tanulás esetén a tanulórendszerhez először adatgyűjtést kell végeznünk. A gyűjtött nyers adatok minősége a végén meghatározza az elérhető pontosságot, ezért fontos a megfelelő minőségű adatoknak a beszerzése. Az adat előfeldolgozása során végezhetünk adattisztítást, normalizálást, átalakítást, jellemzők kiválasztását. Az előfeldolgozás végére kapjuk meg azt a tanítóhalmazt, amellyel a kiválasztott tanulóalgoritmus betanítható. Az adatelemzéshez képest a multimédia-bányászatot nagyobb komplexitás jellemzi. Ennek több oka van: nagy adatmennyiség, az adat nagy változatossága, illetve a strukturálatlanság, és további nehezítés még a multimédia tartalmának szubjektív értelmezhetősége. A jellemzőkinyerést megnehezíti továbbá a multimédia-adathalmazok nagy mérete és magas dimenzionalitása is. A gépi tanulással – a multimédia-tartalmak elemzésében betöltött jelentős szerepe miatt – külön fejezetben foglalkozunk.

Vákát

2. Gépi tanulás a multimédia-elemzéshez

A multimédia-tartalmak elemzését az adatbányászat terminológiát használva hívhatjuk multimédia-bányászatnak is. [Vijayarani–Sakila 2015] A mesterséges intelligenciának (azon belül a gépi tanulásnak) van közös területe, vannak közös jellemzői a multimédia-bányászattal, hisz mindkét esetben a nagy adatkészleteken belüli mintázatok, összefüggések és mélyebb kapcsolatok keresése a cél az eredmények jóslásához. A gépi tanulás algoritmusai adatvezéreltek, így nagy mennyiségű adattal foglalkoznak; és nagy előnyük, hogy rugalmasan képesek megtanulni az adatokban rejlő szabályszerűségeket.

2.1. A gépi tanulás típusai

Háromféle gépi tanulást különböztethetünk meg, amelyek a következők: a felügyelt, a felügyelet nélküli tanulás (más néven nem felügyelt tanulás) és a megerősítéses tanulás *(reinforcement learning).* [Michalski et al. 2013] A felügyelt tanulás esetén összetartozó be- és kimeneti értékek, az úgynevezett tanító mintapontpárok (tanuló állomány) állnak rendelkezésre. A tanítás ezeken az ismert összerendelt mintapárokon alapul. Itt a feladat megtanulni a mintapontpárok által reprezentált bemenet-kimenet leképezést. Mivel ismertek a kívánt válaszok (a bemenetekhez tartozó kimenetek), a tényleges válasz minden esetben közvetlenül összehasonlítható a kívánt válasszal; és a cél a minél kisebb különbség elérése. A nem felügyelt tanulásnál nem állnak rendelkezésünkre adott bemenetekhez tartozó kívánt válaszok, csupán az adatok alapján kell valamilyen viselkedést kialakítania, itt a környezetből nincs semmilyen visszajelzés, ami a viselkedés helyességére utalna. Itt igazából azt kell felderíteni, hogy van-e az adatokban, jelekben valami hasonlóság, van-e az adatok között összefüggés, kialakíthatók-e a bemeneti adatok között különböző kategóriák, csoportok. A felügyelt tanulást akkor tudjuk használni, ha adott bemenetekhez tartozó kívánt válaszok pontos, helyes értékei rendelkezésre állnak. Azonban nem minden esetben rendelkezünk ezen információval, néha csak annyit tudunk, hogy adott bemenetekre a tényleges válasz helyes vagy hibás; tehát a környezetből csak 1 bitnyi visszacsatolt

információ áll rendelkezésünkre. A tanulási eljárás során ez az információ csak arra elegendő, hogy eldöntsük: szükség van-e a tanult modell módosítására vagy sem, de a módosítás mértékének meghatározására már nem elegendő. A tanulásnak ezt a fajtáját megerősítéses tanulásnak nevezzük, megkülönböztetve a felügyelt tanulástól. A kameraképeken, videóstreameken látható érzelem elemzése felügyelt tanulási problémához tartozik, ugyanis az érzelmek előre definiáltak és az adott arcokhoz tartozó kívánt válaszok pontos értékei rendelkezésre állnak.

A multimédia-bányászati módszereket alapvetően több típusba sorolhatjuk: osztályozás, klaszterezés és regresszió. Az osztályozásban és klaszterezésben a közös tulajdonság, hogy mindketten elemek csoportjaival foglalkoznak, de amíg az osztályozásnál ismertek a csoportok (itt felügyelt tanulást alkalmazunk), addig a klaszterezésnél a csoportok nem ismertek, azok meghatározása is feladat (ez a nem felügyelt tanuláshoz tartozik). A regresszió az osztályozással együtt a felügyelt tanuláshoz tartozik, azonban amíg az osztályozó algoritmusok diszkrét értékkészletű célváltozót használnak, addig a regresszió során a célváltozó értékkészlete folytonos.

Az osztályozó algoritmusokat két csoportra bonthatjuk. Az egyik csoport a lusta tanulók, a másik pedig a modellt építők csoportja. Lusta tanulókhoz soroljuk a legközelebbi szomszéd modelleket (k-NN – *k Nearest Neighbour*). Ezek a módszerek nem építenek fel valódi modelleket, az osztályozást csak azután végzik el, hogy az osztályozandó adatot megkapták. A tanító fázisban rövidebb a számításiidő-igényük, azonban a klasszifikáció hosszabb ideig tart. A többi algoritmus ezzel szemben a rendelkezésre álló adatok alapján építi fel modelljeit. A leggyakrabban alkalmazott módszerek, amelyeket idesorolhatunk, a következők: döntési fák, mesterséges neurális hálózatok, naiv Bayes-módszer, Bayes-hálózatok, SVM (*Support Vector Machine* – szupportvektorgépek). Ezek a módszerek legalább két lépcsőben működnek: először modellt építenek a tanítóhalmaz segítségével, a második lépcsőben pedig a felépített modellt alkalmazzák az ismeretlen célváltozójú adatokra, multimédia-tartalmakra.

A regressziós modellek folytonos változók értékének jóslására alkalmazhatók. Ezek közül a legegyszerűbb a lineáris regressziós modell, amely feltételezi, hogy a magyarázóváltozók és a célváltozók között lineáris a kapcsolat. A predikció során egy egyenest próbálunk illeszteni a tanulóhalmaz adatainak pontfelhőjére. A lineáris regresszió becslése során a modell paramétereit becsüljük a rendelkezésre álló mintahalmazból úgy, hogy az az

átlagos négyzetes hibát minimalizálja, ahol a hibán a valódi és a becsült érték közti különbséget értjük.

A gépi tanuláshoz köthető multimédia-elemzésnél több ponton is előfordul a rejtett Markov-modell *(Hidden Markov Model: HMM),* amely annyiban különbözik egy sima Markov-lánctól, hogy az állapotai láthatatlanok, azaz megfigyelők számára „rejtettek". Nem tudjuk, hogy a modell mikor melyik állapotában van, és a köztük lévő állapotátmeneti valószínűségeket sem ismerjük. A HMM-ek jelentősége miatt külön alfejezetben mutatjuk be ennek a modellnek a lényegét és a tanítását.

2.2. A rejtett Markov-modell tanítása

A rejtett Markov-modelleket (HMM) nagyon sok tudományterületen hasznosítják, többnyire sikerrel. Elsősorban beszédfelismerők, akusztikai rendszerek és biológiai algoritmusok esetében alkalmazzák őket, például fehérjekapcsolódások sorrendjének megjósolására vagy beszélőfüggetlen, hangolható beszédfelismerők tervezésére. A HMM lényege, hogy egy – általunk megfigyelhető – folyamatról, megfigyelések sorozatáról (okozat) el tudja dönteni, hogy mi okozhatta azt a legnagyobb valószínűséggel (mi volt az ok?). A módszer statisztikán alapul, és nagy erőssége, hogy tanítható, így megfelelő mennyiségű és minőségű tanítóhalmazzal a hatékonysága egyre javítható.

Sok esetben, mint például a videótartalmakat feldolgozó rendszerek esetében is, a megfigyelhető jelenségek nem véges számú elemmel rendelkeznek, hanem folytonos értékeket vehetnek fel. Az ily módon értelmezett modell a folytonos HMM *(CDHMM – Continuous Density HMM)* [Hernando 1997], ahol az egyes állapotokhoz nem egy-egy valószínűség tartozik, hanem egy eloszlás, azaz hogy az adott állapotban a megfigyelések értéke milyen eloszlást követ. A megfigyelések eloszlására leggyakrabban a normális eloszlás feltételezhető, azaz Gauss-eloszlással modellezhetjük az állapotokon megfigyelhető értékeket. Ebben az esetben az állapotok kimeneti valószínűségeit leíró valószínűség-értékek helyett minden egyes állapotra az ott található Gauss-eloszlások várható értékeit és szórásnégyzet-értékeit adjuk meg, ez fogja jellemezni a modellt.

A modell minden állapotához tartozik tehát egy-egy Gauss-eloszlás, ami az adott állapotot leíró jellemzőkből áll össze a már említett módon. A modell így tehát felállítható, azonban a paraméterek megadása nélkül

nem használható. Egy folytonos HMM-nek a következő három paraméter a legfontosabb alaptulajdonsága:

- *A:* állapotátmeneti valószínűségi mátrix – ez adja meg a belső Markov-lánc állapotátmeneti valószínűségeit.
- *G: N* db Gauss-eloszlás (várható értékekkel és szórásokkal megadva), ahol *N* az állapotok számát jelenti.
- *P: (p1, p2, ... pN),* ahol *N* ugyancsak az állapotok száma – ez az induló valószínűségeket adja meg, azaz melyik állapotból milyen valószínűséggel indul a modell; a valószínűségek összege pedig 1.

A fenti paraméterek természetesen előre nem tudhatók pontosan; az értéküket szeretnénk minél pontosabban úgy megbecsülni, hogy a videók tartalmát legjobban leírja. Az első feladat tehát a paraméterek megfelelő hangolása, tehát az algoritmus tanítása.

Ahhoz, hogy a HMM-et alkalmazni tudjuk, először meg kell állapítani a paramétereit, lehetőleg minél pontosabban. Ezek egyrészt az állapotok közötti éleket jellemző valószínűségek, azaz az A mátrix, másrészt az állapotok jellemzőeloszlásait modellező normál eloszlások várható értékei és szórásparaméterei. Ezeket a paramétereket a legjobban tanítás útján tudhatjuk meg, és a tanulóalgoritmusok nagy előnye, hogy általuk egyre pontosabb becslések kaphatók.

A tanító halmazt, amely jelen esetben videók halmazát jelenti, manuálisan felcímkézzük a következő módon: a videófolyam egyes szakaszaihoz címkéket rendelünk, amelyeknek lehetséges értékei megegyeznek a belső Markov-lánc lehetséges állapotaival. A megjelölés történhet bármilyen módon, például valamilyen videóhoz csatolható metaadat-leíró nyelv segítségével. Ezek után következik a felügyelt tanítás. Az egyes állapotokon lévő többdimenziós Gauss-eloszlások átlag- és varianciaparamétereinek jó meghatározását szolgáltatja a következő módszer.

Először kinyerjük a felcímkézett tanító videókból a jellemzővektor-sorozatokat, alapesetben minden képkockáról (de lehet mintavételezni minden *n*-edik képkockát is). Mivel rendelkezésre állnak a videók címkéi, és ezek megjelenési idejét össze lehet egyeztetni a kinyert jellemzők képkockájának (például milliszekundumban) mért idejével, így tudható, hogy az egyes állapotok történése alatt milyen jellemzővektorokat figyeltük meg. Minden egyes állapotra, a neki megfelelő vektorokat megadva, az ott lévő Gauss-eloszlás paraméterei jól megtaníthatók az *Expectation-Maximalization* (EM) módszercsalád [Moon 1996] segítségével.

Tanításnál leggyakrabban a Baum–Welch [Wu et al. 1999] algoritmust alkalmazzák, ez a HMM-ek paramétereinek beállítására használt módszer az EM-algoritmusoknak egy fajtája, *forward* és *backward* algoritmusok kombinációja. Mindez dinamikus programozással valósítható meg, és képes a HMM-paraméterek hangolására *(smoothing),* csupán a beadott megfigyelésvektor-sorozatok segítségével.

Miután a fenti lépések, azaz a tanítás megtörtént, egy jól használható modellt kaptunk, amely alkalmas videók elemzésére (például mozgásdetekcióra vagy az állapotoknak megfelelő részekre való szegmentálásra). Az állapotok egymás után következése tehát lépések sorozata a Markov-láncon, a modell valószínűségek segítségével reprezentál minden lehetséges előálló állapotot. Ahhoz, hogy megtudjuk, hogy egy adott videóban hogyan következtek ezek egymás után, azaz hogyan jártuk be a Markov-láncot, kell a megfigyelés a videóról és egy módszer, amely ezen megfigyelés alapján megmondja, hogy milyen út bejárásával jutottunk ezen megfigyeléshez.

Egy ilyen módszer a Viterbi-algoritmus [Forney 1973], amely alkalmazható nemcsak HMM-ek tanítására, hanem kiértékelésükre is. Általános működése a következő, ami egyben a rejtett Markov-modell használatának és kiértékelésének lényege is. Egy megfigyeléssorozathoz megadja, hogy melyik az a legvalószínűbb mód, ahogyan ehhez a megfigyeléssorozathoz eljutottunk. A Markov-modellre lefordítva a működése: egy adott videóból kinyert képjellemzővektor-sorozatra megmondja, hogy a Markov-láncban melyik a legvalószínűbb állapot-lépéssorozat (Viterbi-útvonal), amelyből a képjellemző-sorozatot kaptuk; egy úgynevezett Trellis-gráfot [Pye et al. 1998] kapunk eredményül a bejárásról. Utolsó lépésként a Viterbi által visszakapott állapotbejárás-sorozatot rá kell illeszteni az azt generáló jellemzővektorok sorozatára. Ezt már a kinyeréskor úgy tároljuk el, hogy minden jellemzővektorhoz eltároljuk a képkockájának a videóban való, milliszekundumban mért megjelenési idejét is. Így a visszakapott útbejárási sorozat mellé azt is megtudjuk, hogy mely időpillanatokban, mely állapotban tartózkodott a videófolyam, vagyis a HMM.

2.3. Párhuzamos architektúrák a gépi tanulásban

A számítástechnika (azon belül is elsősorban a GPU: Graphics Processing Unit) által nyújtott párhuzamos architektúra lehetősége nagyban megnövelte a gépi tanulások hatékonyságát. Az új architektúrák lehetőségeinél a cél

ugyanaz maradt, azaz a számítógépeknek úgy kell viselkedniük, mint egy emberi lénynek, azzal a különbséggel, hogy kevesebb technikai korlátja van. Ennek köszönhetően a fejlődés menete gyors volt, ezt mutatják az elmúlt évekből kiragadott kiemelkedő eredmények is:

- 2015-ben a Google cég megjelentette a GoogLeNet-et [Szegedy et al. 2015], majd egy évvel később az InceptionV3-at [Szegedy et al. 2016], amely a konvolúciós neurális hálózat harmadik kiadása volt. Ez a hálózat a képek pontosabb besorolására képes, mint az emberek.
- 2016-ban az AlphaGo, amelyet az Alphabet Inc., a Google DeepMind számítógépes csoportja fejlesztett ki [AlphaGo 2016], és amely (*a számítógépek között*) először bizonyult jobbnak, mint a GO világbajnoka.
- Számos mesterségesintelligencia-kutató szerint a 2018-as év legnagyobb innovációja egy olyan chatbot lehet, amely átmegy a Turing-teszten, így nem fogjuk észrevenni, hogy egy géphez beszélünk. Ebben az érzelemfelismerés is nagyon fontos szerepet tölthet be.

Az ember és a számítógép összevetéséből adódóan több tudós is próbálkozott azzal, hogy a számítógépeket úgy alakítsák, hogy azok jobban hasonlítsanak a párhuzamos architektúrájú agyra. Ezt a neurális hálózatok segítségével érték el, amelyek több száz, ezer vagy millió mesterséges „agysejtből" összeállított számítógépes programok, amelyek az emberi agyhoz hasonlóan tanulnak és viselkednek. Pontosabban a mesterséges neurális hálózat *(Artificial Neural Networks: ANN)* egy olyan számítási modell, amelyet a biológiai neurális hálózatok emberi agyi folyamatokkal kapcsolatos információi inspiráltak. Természetesen van lényeges szerkezeti különbség a számítógép és az agy között: az agyban az agysejtek gazdagabb és bonyolultabb módon kapcsolódnak össze. A számítógépeket arra tervezték, hogy óriási mennyiségű (értelmetlen) információt kapjanak, és ezeket sokféleképpen átrendezzék azoknak a pontos utasításoknak (programoknak) megfelelően, amelyeket előzetesen betöltünk a gépbe. Az agy viszont lassan tanul, inkább körkörös módszerrel, gyakran hónapok vagy évek alatt, hogy teljesen megértsen egy nagyon összetett problémát. A mesterséges neurális hálózatok számítógépek programozásával készülnek, és úgy viselkednek, mintha milliárd párhuzamosan működő, erősen összekapcsolt agysejtekből építették volna őket. Más szavakkal a mesterséges neurális hálózat különbözik az emberi agytól ugyanúgy, mint az időjárás számítógépes modellje a valódi időjárástól. A mesterséges neurális hálózatok fejlődése a gépi tanulás kutatásában főleg

az iparnak köszönhető, hiszen olyan alkalmazási területeken kellett gyors és lehetőleg áttörő eredményeket elérni, mint a beszédfelismerés, a számítógépes látás és a szövegfeldolgozás; és ezek mindegyikénél alkalmazhatók voltak azok a megoldási struktúrák, amelyekkel a neurális hálózatok alapból rendelkeznek. Ezeket a neurális hálózatokat lehet a GPU segítségével gyorsan tanítani és alkalmazni, aminek sikerességét az (utóbbi néhány évből) bemutatott példák igazolják.

Vákát

3. A kamera képeinek feldolgozása

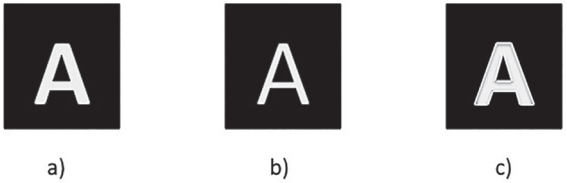
A számítástechnikában a képfeldolgozás alatt egy olyan jelfeldolgozást értünk, amikor a bemenet egy kép (amely lehet valamilyen fotó vagy egy kamerakép) és a kimeneten megkapjuk a kép jellemzőinek egy halmazát (esetleg a kimenet lehet egy másik kép is). A legelterjedtebb módszerek a képet kétdimenziós jelként dolgozzák fel – ez főleg a szürkeárnyalatos képekre igaz. A színes képek több csatornával (három színcsatornával) rendelkeznek, és emiatt a több kétdimenziós jel felfogható háromdimenziósként is, ahol a harmadik dimenzió annyi értéket vehet fel, amennyi a csatornaszám. A képek rendelkeznek olyan tulajdonságokkal, amilyenek jól jellemezhetik a képet, segíthetik a további feldolgozási lépéseket. Az ilyen tulajdonságok például a kép intenzitása, a kép színei, a képen megjelenő élek, textúrák.

3.1. Morfológiai képfeldolgozás

A képfeldolgozást az alacsony szintű műveletek bemutatásával kezdjük. Többek között idetartoznak a morfológiai képfeldolgozással kapcsolatos eljárások. [Najman–Talbot 2013] Ezen eljárások alapja az erózió-, illetve a dilatációművelet. Ezek segítségével kiszűrhetjük a képből a zajokat, illetve elkülöníthetjük az egyes objektumokat. Elsősorban bináris, kétszínű képeken használják őket. Mindkét műveletnél szükség van egy úgynevezett strukturáló elemre. Ez egy bináris mátrix, amellyel egy adott pontban megvizsgálhatjuk az illeszkedés mértékét.

Az erózió esetében az adott pont akkor lesz fehér, ha a strukturáló elem illeszkedése minden pontban tökéletes. Ha nincs teljes fedés, akkor az a képpont fekete lesz. Dilatáció esetén a helyzet fordított, csak akkor lesz a képpont fekete, ha a strukturáló elem minden pontjában fekete a kép adott pontja – részleges fedés esetén tehát a pont fehér lesz. A 4. ábrán látható a két művelet hatása: a bal oldali, a) részen az eredeti objektumot mutatja, amelyen elvégezve az erózió műveletét a középső b) kis képet kapjuk, illetve a jobb oldalon, a c) részen látható a dilatáció hatása.

A két művelet együttes (pontosabban közvetlenül egymás utáni) alkalmazásával fontos képmódosítási eljárásokat tudunk megvalósítani. Először a dilatáció, majd az erózió művelete – amelyet zárásnak hívunk – segítségével eltüntethetjük az objektumon belüli üres részeket. A nyitás nevű eljárás az ellentétes sorrenddel áll elő, amikor is az erózió műveletével kezdünk: ezzel azokat a zajos részeket tüntethetjük el, amelyek olyan kicsik, hogy nem tekinthetők objektumnak.



4. ábra

Morfológiai műveletek: a) eredeti objektum; b) erózió után és c) dilatáció után

Forrás: a szerzők szerkesztése

3.2. Képi hisztogram

A képi hisztogram szintén az alacsony szintű képfeldolgozáshoz tartozik (statisztikai megközelítéssel), amely egy olyan hisztogram, amely egy digitális kép árnyalatainak felosztását reprezentálja megjelenítve azt, hogy az adott árnyalathoz hány pixel tartozik. Ezt a fajta grafikont megnézve képet kaphatunk az egész kép árnyalatainak felosztásáról. A képi hisztogram megjelenítése mára már majdnem minden digitális kamerán megtalálható, mivel ezzel a fényképészek könnyen ellenőrizhetik az elkészített képet. A grafikonon egyből láthatóvá válik, ha a kép túl világos lett, vagy ellenkezőleg: túl sok, részlettelenül sötét rész van a képen.

A képfeldolgozásban a képi hisztogramok azért hasznosak, mert az általuk hordozott információk segítségével fontos megállapítások tehetők, és segíthetik a képfeldolgozási módszereket. Egy hisztogramon megkeresve a csúcsokat, illetve a völgyeket, könnyen kaphatunk megfelelő küszöbértékeket a további képfeldolgozási lépésekhez. Ilyen küszöbérték szükséges például az éldetektáláshoz is.

A közismert RGB-színábrázolás helyett a hatékonyabbnak tartott HSV [SURAL et al. 2002] színábrázolást érdemes választani, hisz ez a színek reprezentálását jóval „emberközelibb” módon oldja meg. A piros, zöld és kék színek összeadása helyett a képek színezetét *(H – hue),* telítettségét *(S – saturation)* és világosságát *(V – value)* kódolja, ezekből adódik a három összetevő.

A H, S és V értékek egy adott képen önmagukban is hordoznak jelentést, nemcsak álló-, hanem mozgóképek esetén is. Például egymást követő képkockákat tekintve, fokozatosan csökkenő világosságértéket *(value)* észlelve valószínűsíthető, hogy halványításeffekt *(fade)* van folyamatban. Ez sokszor egyben egyféle jelenethatárt is jelent a videók esetében, amit a szegmentáló programnak meg kell találnia, így érdemes az egyesített, tömörített hisztograminformáción kívül a színábrázolás három összetevőjét külön-külön is számításba venni. A H, S és V értékek egy kép esetén egy-egy eloszlásnak tekinthetők annak minden tulajdonságával együtt. Mivel a képkockákban levő pixelek száma nagy, ezért azok tulajdonságai, az eloszlások jól jellemezhetők azok statisztikai jellemzőivel. Minden képkockára, mind a három (H, S, V) síkon kiszámítható azok átlagértéke és varianciája. Míg az előbbi az adott sík átlagolt értékét (például alacsony átlag V esetén sötét kép) jelenti, az utóbbi annak szórását jellemzi jól (például magas szórású H esetén színpompás kép). Így a hisztogramtávolság mellett további hat jellemző írható fel: három átlagérték H, S és V értékekre és három másik jellemző: a H, S és V varianciái.

3.3. Éldetektálás képeken

A kamera képeinek feldolgozásánál az egyik legalacsonyabb szintű művelet az éldetektálás, amely a képfeldolgozás alapvető eszközei közé tartozik. A segítségével megállapíthatók a képen a meredek intenzitásváltozások. Ennek segítségével kaphatunk egy kevesebb információt tartalmazó képet, amely csak a kinyert éleket tartalmazza. Ezt a képet könnyebb feldolgozni, és az élek miatt nem veszíti el fontos tulajdonságait, struktúráját.

Az éldetektálás nehézsége abban áll, hogy nehéz megválasztani azt a küszöbértéket, amikor már egy intenzitásváltozást élnek tekintünk. Ez sok mindentől függhet, mert éleket több különböző tényező is létrehozhat. Élként jelenhet meg például a képen a szín intenzitásának változása, a mélység vagy a megvilágítás. A küszöbérték rossz megválasztása vagy túl sok élt eredményez, ami nehezíti a további feldolgozást, vagy lehet éppen túl kevés

él, amikor is hasznos információk vesznek el a képből, amelyek később feldolgozhatók lettek volna.

A Canny-éldetektor [Canny 1986] az egyik leghatékonyabb és legelterjedtebb ilyen algoritmus az élek megtalálására. Az algoritmus három kritériumnak igyekszik megfelelni: csak létező éleket találjon meg, az éleket pontosan lokalizálja, minden élt pontosan egyszer jelezzen. Az algoritmus négy fő lépésből áll:

- a képzaj szűrése, képsimítás Gauss-szűrővel,
- gradiensek becslése, az intenzitásgradiens nagyságának és irányának meghatározása,
- nem az élhez tartozó pontok elnyomása,
- hiszterézis segítségével (küszöböléssel) a megtört élek összekötése.

Ezt a négy lépést az alábbiakban részletezzük:

1. A képzaj szűrése, képsimítás Gauss-szűrővel. Minél nagyobb szűrőt használunk, az élek annál jobban elkenődnek, csak a fontosabb, nagyobb méretű éleket fogja detektálni az algoritmus. Nagyobb szűrő esetén azonban az algoritmus kevésbé pontosan tudja kiszámolni egy él helyét. Kisméretű szűrővel az apróbb részletek is megmaradnak, így a detektor meg tudja találni őket.
2. Gradiensek becslése, az intenzitásgradiens nagyságának és irányának meghatározása. Mivel a képek kétdimenziós függvényként foghatók fel (x és y irányban), ahol az intenzitás képpontról képpontra változhat, ezért lehet az intenzitásváltozást vizsgálni mind annak nagysága, mind pedig iránya szempontjából.
3. Nem az élhez tartozó pontok elnyomása. Az élek ott vannak, ahol a gradiensnek lokális maximuma van: ami az élre merőleges (a gradiens irányába esik), azt el kell hagyni.
4. Hiszterézis segítségével (küszöböléssel) a megtört élek összekötése. A Canny-algoritmus két küszöbértéket használ, mert ha az élek értéke egy küszöb körül ingadozik, akkor sok szakadás keletkezik. Ezért egy alsó és egy felső küszöböt kell kijelölni. Ha a vizsgált pixel gradiense nagyobb, mint a felső küszöb, akkor felvesszük élnek. Ha kisebb, mint az alsó küszöb, akkor eldobjuk. Ha a két küszöb közé esik a gradiens, akkor csak abban az esetben vesszük fel élnek, ha egy szomszédos pixel már egy élhez tartozik.

Az éldetektorok kimenetét használhatjuk a képtelítettség leírására. A megtalált élek az eredeti vizsgált képpel megegyező méretű fekete képre rajzolhatók fehérrel, majd ezen a képen morfológiai zárással egy olyan képet kaphatunk, amelyen az egymáshoz közel levő élekből egy vastagabb vonal lesz. Ezzel a módszerrel elérjük azt, hogy egy olyan kép áll elő, amelyen a fehér pixelek száma arányos a vizsgált képen levő objektumok (tárgyak) számával. Ha a vizsgált kép homogén, azaz kevésbé telített, akkor kevés él detektálható. Ha azonban a kép zsúfolt, sok tárgy van rajta, akkor a sok intenzitásváltásnál sok él látható, amelyek a morfológiai zárás során még jelentősebb, nagyobb arányt képezhetnek. Az éldetektálás előtt a képet érdemes egy Gauss-szűrőn átengedni, hogy az apróbb zajszerű intenzitáskülönbségeket ne vegye élnek a detektor. A morfológiai zárás során a dilatáció és erózió műveletéhez is egy kör vagy ellipszis alakú strukturáló elemet érdemes hozzárendelni. A képtelítettségre számolt végső jellemzőként a morfológiai zárás után az elkészített képen össze lehet számolni a fehér pixelek számát, majd ezt arányosítani az összes (fekete és fehér) pixelszámhoz, ezzel a normalizálással megkaphatjuk a képtelítettséget. Egy másik járható út a képtelítettség mérésére a kép szegmentációján alapul. A képszegmentáció alatt értjük a teljes képnek kisebb homogén részekre való felbontását. Ekkor a képszegmensek száma szintén jól tudja mutatni a képtelítettség mértékét.

3.4. SIFT-jellemzők a képeken

A SIFT- *(Scale Invariant Feature Transform)* algoritmus segítségével egy forrásképen megjelenő objektumot ismerhetünk fel egy másik képen. [Lowe 2004] A felismerendő objektumnál olyan kulcsfontosságú pontokat érdemes kiragadni, amelyek lehetővé teszik az objektum „jellemző leírását”. Ezt a leírást a forrásképből kinyerve használhatjuk arra, hogy azonosítsuk az objektumot egy tesztképen (azaz megtaláljuk a keresett objektumot a többi között). Ahhoz, hogy megbízható felismerést érjünk el, a forrásképből kinyert jellemzőknek detektálhatónak kell maradniuk a kép méretének, zajszintjének vagy megvilágításának megváltozása esetén is. Ezeket a pontokat általában a magas kontrasztú területeken találjuk, ilyenek például az objektum élei, sarkai. A másik fontos elvárás ezekkel a jellemzőkkel szemben, hogy a relatív helyzetük nem szabad, hogy megváltozzon az eredeti kép (forráskép) és a tesztkép között (ha egyik képről a másikra a pontok relatív helyzete megmarad, akkor detektálható lesz az objektum az új képen is;

máskülönben, ha bármilyen változás történik a két kép között az objektum belső geometriájában, akkor a felismerés valószínűleg nem fog működni). A gyakorlatban a SIFT-algoritmus nagyszámú jellemzőt készít a képekről, ami csökkenti a lokális változásoknak az egész felismeréshez hozzáadott hibáját. Lowe szabadalmazott módszere tehát robusztus megoldást kínál objektumok azonosítására, legyenek azok zsúfoltan vagy csak részlegesen láthatók egy képen. Ennek oka az, hogy a SIFT-jellemzők (leírók) nem érzékenyek a méretváltozásra, orientációváltozásra és az affin torzulás esetére, valamint részlegesen érzékenyek csak a megvilágítás változás esetére.

A SIFT-algoritmus a következő négy lépésben állítja elő a SIFT-jellemzőket (leírókat).

Extrém pontok keresése a skálatérben. Az extrém pont olyan szélsőértékkel rendelkező pont, amely kicsinyítésnél és nagyításnál is szélsőértékkel rendelkezik. Ennek a megvizsgálásához a vizsgált képhez felépítünk egy többszintű skálateret, azaz minden szinten előállítjuk a képnek egy transzformációját (például kicsinyítéssel). Ebben a skálatérben keresünk skálainvariáns pontokat úgy, hogy a kép egy adott pontjához stabil jellemzőket keresünk minden lehetséges skálán. Ezeket ott fogjuk megtalálni, ahol a két szomszédos skálaszint különbségéből származtatott képen (az úgynevezett különbségképen) a pixelintenzitásokban szélsőértéket találunk. Ezek lesznek az extrém pontok.

Kulcspont-lokalizáció. Miután kiválasztottuk a jelöltjeinket (az extrém pontokat) a szomszédjaikkal való összehasonlítás után, a következő lépés a közelben lévő pontok megvizsgálása elhelyezkedés, skála, valamint elsődleges deriváltak szempontjából. Az extrém pont környezetében található pontok intenzitásértékeinek (függvényként tekintve őket) deriváltjaiból kiszámítható a vizsgált pont kontrasztja. Ezzel az alacsony kontrasztú pontokat visszautasíthatjuk, mivel ezek érzékenyebbek a zajra. A maradék extrém pontok lesznek a kulcspontok.

Orientáció kijelölése. Amennyiben mindegyik kulcsponthoz hozzárendelünk egy egyértelmű orientációt a kép helyi tulajdonságai alapján, akkor a leírót ennek megfelelően tudjuk reprezentálni, amivel elforgatásinvarianciát érhetünk el. A kulcspont körüli régió egyes irányokba eső gradienseit kiszámolva kiválaszthatjuk azt az egy fő orientációt, ami a kulcspont orientációja lesz.

Lokális képleírás a kulcspontokhoz. Az előző lépésekkel megkerestük az extrém pontok közül a kulcspontokat, és mindegyikhez orientációt rendeltünk. Ebben az utolsó lépésben egy robusztus leíró megalkotása a cél,

amelyet úgy alkothatunk meg, hogy megvizsgáljuk a kulcspont környezetét. A környező pontok orientációi egy mintázatot adnak, és ez a mintázat által meghatározott számsorozat lesz a végleges leíró.

Vákát

4. Videótartalmak rendszerezése

Sok kutatás foglalkozik videórendszerezési, -elemzési témákkal, rendkívül érdekes módszerekkel, javaslatokkal és keretrendszerekkel, néhol gyakorlati teszteredményeket is felmutatva. Azonban a valós életben – a feladatok összetettsége és sokrétűsége miatt – kevés az olyan használható megoldás, amely átfogó módon sok funkciót meg tud valósítani. A probléma sok oldalról megközelíthető, és sok esetben, például képek csoportokra bontása vagy videók jelenetekre vágása esetén a szubjektív döntés miatt sosem lesz globálisan, mindenki által elfogadott, tökéletes megoldás a kérdésre, azonban lehet törekedni a minél jobb eredményekre.

4.1. Csoportosítási szempontok

A videók rendszerezése és elemzése sokféle módon, többféle szinten és szempont szerint történhet – a céltól függően. Lehet a mozgóképek szemantikus indexelésére és szegmensekre bontására helyezni a hangsúlyt, így kereshetővé téve azok tartalmát. Ezután az indexeléshez kinyert információk alapján a feldolgozott videók halmazát akár többféle különböző csoportba is rendezhetjük. A továbbiakban bemutatjuk a mozgóképek kezelésének főbb aspektusait és módszereit, megvilágítjuk a probléma sokrétűségét, annak kihívásait és a megoldás lehetőségeit.

A videótartalmak kategóriákba sorolása többféle megközelítés szerint történhet, itt felsorolunk néhány szempontot, amelyek alapján a besorolás végezhető.

- Cél(közönség) szerint: felnőttfilmek, gyerekműsorok, oktató videók stb. Ez a videók legalapvetőbb célját, rendeltetését határozza meg.
- Műfaj, alműfaj szerint: a műfaji besorolásokhoz számos iparági szabvány, illetve ajánlás létezik. Ezek legtöbbször egy hierarchikus felbontást jelentenek, tehát egy mozgóképről megmondható például, hogy felnőtteknek készült, ezen belül hírműsor, még specifikusabban politikai hírműsor. Egy előre elkészített hierarchikus csoportosítás

bármilyen tartalomhalmaz esetén alkalmazható, ám elképzelhető automatikus kategóriákba sorolás (osztályozás) is.

A videók lehetnek nagyon sokféle típusúak, érkezhetnek eltérő forrásokból. Ha éppen film típusú videókról készítjük az elemzést, akkor ebben a specifikus esetben a film műfaj szerinti besorolása is egy értékes attribútum. Ennek a műfajattribútumnak a lehetséges értékei egy angol filmes adatbázis alapján: *action, adventure, animation, biography, comedy, crime, documentary, drama, family, fantasy, film-noir, game-show, history, horror, music, musical, mystery, news, reality-tv, romance, sci-fi, sport, talk-show, thriller, war, western.*

Akár egy kamera által folytonosan vett felvételben, akár egy kész videótartalomban az automatikus jelenetfelismerés hasznos lehet az okosváros-projektek szempontjából. Ehhez a felismeréshez különböző, alacsony szintű jellemzők által szolgáltatott, feldolgozott információk szükségesek. Például egy sportesemény, azon belül is egy teniszmérkőzés közvetítése során néhány jól meghatározható kameraállás, színek és képi világ definiálja az egyes jelenettípusokat: szerva, ász, szett stb. A teniszmeccset tartalmazó videók esetében is a videóból kinyert (nyers) képadatokat, képkockákat különböző képmanipuláló és -elemző módszerekkel kell feldolgozni, hogy az automatikus jelenetfelismeréshez eljussunk. Mivel tapasztalatok alapján ismerjük a teniszmeccsek menetét (például ha a kép feléig majdnem csak homogén zöld szín van a képen, a tetején egy csík vízszintesen és lent egy kis emberalak, akkor egy szervanyitást látunk), ezért ezt kihasználva a mérkőzés automatikusan elemezhetővé válik, és megadott részekre bontható. A zöld szín meglétét, az emberalakot stb. az alacsony szintű képi feldolgozással detektálni lehet, így ez a módszer egy automatikus megoldást adhat egy-egy konkrét műsortípuson belül a nézőket legjobban érdeklő jelenetek felismeréséhez. Ugyanilyen elven alapulva lehet az utcai térfigyelő kamerák esetében is automatikus jelenetdetekciót megvalósítani például egy utcai verekedés vagy más atrocitás esetében.

Az elemzéshez (a képi jellemzők kinyeréséhez) a mozgóképet alapegységekre kell bontani, ami során a felvétel videóképkockákra *(frame)* bontható valamilyen fps- *(frames per second)* felbontással. A *frame-ekre* bontás során minél nagyobbra állítjuk az fps-értéket, természetesen annál több *frame-et* nyerünk ki a videóból, így finomabb felbontást kapunk, és elemezhetők a hirtelen történő változások, mozdulatok is. Viszont a több *frame* jóval több számítási kapacitást is jelent, mivel mindegyikre ki kell számolni

a képi jellemzőket. Így érdemes egyensúlyt találni a számítási kapacitás és a felbontás finomsága között.

4.2. Multimodalitás kérdése

Az emberi agy rendkívüli feldolgozó- és értelmezőképessége miatt fel sem tűnik, hogy amikor egy videót nézünk, mennyire összetett információhalmazról van szó. Amit itt meg kell említeni, az a multimodalitás. Többek között ez az, ami kiemeli a videókat a többi médiaformátum közül. A multimodális szót használják más területeken is (például emberi érzékelés esetén beszélhetünk taktilis, akusztikus, vizuális modalitásokról), de jelen tanulmányunkban a multimédia-tartalmakra értelmezzük a modalitást [Maragos et al. 2008], azaz azt a módot, ahogyan a fogyasztó azt befogadni képes. Így vannak unimodális típusú tartalmak: szöveg-, audio- és képtípusúak, és vannak multimodálisak, amikor ezekből egyszerre több is jelen van, mint például a karaoke esetében a szöveg és audio, vagy az audiovizuális mozgóképek (például egy feliratos film) esetén mindhárom.

Az audiovizuális mozgóképek kifejezőereje rendkívül nagy, készítőik is szándékosan több módon juttatják el a nézőkhöz a gondolataikat: a képi megjelenítéssel a nézők képzelőerejére lehet hatni, ám a hangokkal az érzelmekre és érzékekre hathatunk (kiterjesztve a képen elmondottakat). A hanghatások néha jobban megérintik az embereket, és érzékenyebbek rá, mint a látványra, így a videók feldolgozásánál is érdemes ezeket számításba venni. Szöveges információk is találhatók a mozgóképeken, amelyek kevésbé jelentősek ugyan, de remek plusz információkat adhatnak a jelentésekhez. Ha a képen kívül elemezni akarjuk a többi modalitást is, akkor szövegek esetén szövegbányászati (nyelvelemzési) metódusok alkalmazhatók, hanganyagok/zenék esetén pedig különböző hangfeldolgozási algoritmusok és módszerek használhatók. Az elmúlt években készült munkák eredményeit tanulmányozva megállapítható egyfajta trend. Az eddigi és a közelmúltban alkalmazott eljárások legtöbbje csak a videók fő komponensére, a képi anyagra koncentrált, és a képi világból próbált következtetni a videók tartalmára, jelentésére, és próbálta így azokat rendszerezni. Azonban amíg ez a módszer állóképek esetén jól működik, mozgóképeknél nem mindig elégséges. A mozgóképek rendkívüli változatossága és összetett mondanivalója valójában sokszor nem fejthető meg csupán képi információk alapján. Így a kutatók egyre inkább a videók multimodális feldolgozására koncentrálnak:

– a videók közben hallható hanganyagot és látható szöveges információkat is beleveszik a rendszerezésbe. Ez természetesen tovább bonyolítja az amúgy sem egyszerű feladatot, viszont cserébe sokkal precízebb, „emberközelibb” eredmények érhetők el.

A multimodális feldolgozás egyik fontos kérdése az elemzésnél, rendszerezésnél, hogy hogyan kombináljuk az egyes módszereket. Az egyik megközelítés szerint a kinyert képi és hang- (esetleg szöveges) jellemzőket együttesen, egy közös algoritmusban (például többdimenziós vektorként mint bemenő paraméter) felhasználva alkalmazzuk. A másik módszer szerint külön-külön végzünk elemzést minden modalitás mentén, majd ezeket utólag kombináljuk valamilyen módon.

Az információk beépülése szempontjából a mozgóképelemzésnek két alapvető módja van: a metaadat-alapú és a tartalomalapú megközelítés. A következő két alfejezetben ezeket mutatjuk meg részletesen.

4.3. Metaadat-alapú megközelítés

A metaadatok, azaz „adatok az adatról” hasznosak az információfeldolgozás minden területén. Ahogy a képekhez például az EXIF *(Exchangeable Image File Format)* vagy mp3-hoz az ID3-tagek, úgy a videókhoz is léteznek leíró formátumok, amelyek az adott videó tartalmát hivatottak részletezni szöveges formában, szabványosított módon. Ennek egyik képviselője az MPEG-7 szabvány [Sikora 2001], ami a közismert MPEG-1, MPEG-2 és MPEG-4 szabványokkal ellentétben nem a konkrét audiovizuális tartalom kódolásáért felelős, hanem annak metaadat-leírásáért. Architekturális alapelvei közt szerepel, hogy a leírást a konkrét tartalomtól el kell különíteni, viszont kell, hogy legyen kapcsolat a kettő között. Erre időkódolást *(timecoding)* használ, vagyis a videó különböző időpillanataihoz is lehet adott eseményeket, leírásokat, például zeneszövegeket rendelni. Tipikus alkalmazás például különböző nyelvű feliratok megadása egy videóhoz. Az MPEG-7 fő részei:

– Leíró definíciós nyelv *(DDL – Description Definition Language):* XML-alapú, a struktúrát alapozza meg.
– Leíró séma *(DS – Description Schemes):* szemantika és kapcsolatok az MPEG-7 elemek között – leírók és leíró sémák között.
– Leíró *(D – Descriptor):* szintaktikai és szemantikai leírása egy adott elemnek, például egy konkrét eseménynek a videóban.

– Rendszereszközök: a leírók szállítását, tárolását, szinkronizálását oldja meg, továbbá szerzői jogok védelmére is alkalmas.

Az elemzés során kinyert információkat érdemes valamilyen módon eltárolni, lehetőleg valamilyen szabványosított, direkt ilyen célokra készült struktúrát használva, például az említett MPEG-7 szabványt. Akár egy külön modul készíthető, amely a kinyert információkat MPEG-7 formátumba konvertálja, így kapcsolva a videóhoz azokat; így egy automatikus és intelligens MPEG-7 generáló/kitöltő alkalmazás nyerhető.

Az MPEG-7 segítségével (ha helyesen vannak kitöltve a metaadatok) könnyedén és gyorsan lehet videókat automatikusan jellemezni, csoportosítani és visszakereshetővé tenni. Konkrét megvalósítások is születtek rá, például a BilVideo-7 [Bastan et al. 2009], amely egy adatbázis-alapú komplett architektúrát használ. Azonban a metaadat-alapú megoldásoknak megvannak a hátrányai is, ami a következő esetekben okoz nehézséget.

– A metaadat nincs kitöltve a videóhoz: a metaadatok jellemzően emberek által, manuálisan kitöltendők, de sok esetben a felhasználók nem veszik a fáradságot, hogy a kitöltsék.
– Ki van töltve a metaadat, de hibásan – ez még nehezebben kezelhető probléma, ugyanis ekkor a videók elemzése rossz adatok alapján történik meg, így rossz eredményeket kaphatunk.
– Nem lehet kitölteni, mint például biztonsági kamerák által közvetített streamvideók esetében.

Ezen okok mutatják, hogy a metaadat-alapú megközelítés bár nagyon effektív és ígéretes, mégsem tekinthető általános megoldásnak, ami minden esetben alkalmazható. Érdemes tehát tartalomalapú megközelítést is használni. A következőkben ezt vizsgáljuk meg részletesen.

4.4. Tartalomalapú megközelítés

A tartalom alapján történő feldolgozás a videók esetén azt jelenti, hogy kódolt bitfolyamból, a videó képekre *(frame-ekre)* bontott halmazából vagy a hangsávból nyerünk ki információt. Erre különböző matematikai módszerek, algoritmusok és technikák léteznek.

4.4.1. Képitartalom-alapú megközelítés

A videó képi tartalmát, látványvilágát, a megtalálható objektumokat elsősorban a *frame-ek* képelemzésével vizsgálhatjuk. Ezekre megfelelő technikák a színintenzitások és színhisztogramok használata, a SIFT- [Lowe 2004], SURF- [Bay et al. 2006] vagy más objektumegyezést, képleíró-egyezést vizsgáló módszerek. Az alapmegközelítés, hogy egymást követő *frame-ek* között valamilyen módon definiált távolságot vizsgálunk, és ha az egy határértéket átlép, akkor egy új képcsoportról (például *shotról,* jelenetről) beszélünk. A határérték megtalálása persze a videók sokszínűsége és a különböző effektek (átmosás, halványodás, kameramozgások, rossz minőségű digitalizált videó) zavaró hatása miatt rendkívül nehéz is lehet, így további kiegészítő módszerek szükségesek.

Képitartalom-alapú megközelítésnél magasabb szintű detekciók állíthatók elő az alacsonyabb szintű képfeldolgozási módszerekből. Így videókban megtalálhatók/detektálhatók különböző vizuális tartalmi entitások, mint például emberek, arcok, egyszerűbb emberalak- vagy összetettebb arcfelismerő technikák segítségével. Emberek detektálása esetén különösen hasznos lehet a videó hangsávjának elemzése is, például beszédfelismerő eljárások: megállapítani, hogy épp ki beszél. Ezen kívül különböző objektumok, tárgyak felismerése is történhet objektumfelismerő, objektumleíró és egyezést vizsgáló módszerek segítségével. A háttér detektálása is nagyon hasznos lehet, hiszen egy *shot* alatt általában ez nagyon keveset változik, így alkalmazható egyfajta közös háttérként az előtér objektumaira, mozgás esetén pedig referenciaként.

4.4.2. Hangalapú megközelítés

A videó hangjának elemzésére számos hangelemző módszer és algoritmus áll rendelkezésre. A különböző típusú hangok csoportosítására elindultak már a kutatások [Zhang–Kuo 1998], amelyek során a hangokból kinyert alacsony szintű információra a ZCR- *(average zero-crossing rate)* módszert alkalmazzák, majd HMM (*Hidden Markov Model* – rejtett Markov-modell [Bourlard–Morgan 1994]) segítségével osztályozzák azokat, amely képes a hangok időbeli változásainak leírására. A [Breebaart–Mckinney 2004] publikációban többek között különböző alacsony szintű hangtulajdonságok (keménység, hangosság, élesség) és Fourier-transzformáció segítségével

nyernek ki hanginformációkat, majd osztályozzák a különféle hangokat megkülönböztetve a popzene, klasszikus zene, beszéd, zaj és tömegzaj osztályokat.

4.4.3. Kombinált megközelítés

A különböző modalitások (főként a vizuális és hangalapú) kombinálása hasznos lehet; sokkal jobban, és nagyobb hatásfokkal/pontosabban elemezhetők ily módon a videótartalmak azáltal, hogy mindkettőnek az előnyeit próbáljuk a kombinálásban kihasználni (kiküszöbölve a hátrányokat). A hangsáv például segítségünkre lehet különböző, a képfeldolgozást megzavaró effektek (például átúsztatás, zoomolás stb.) esetén. Egy ilyen kombinált megoldás [Boreczky–Wilcox 1998] kidolgozói a *shotok* megtalálására tesznek kísérletet, méghozzá bimodális hozzáállással, a videó hanganyagát is felhasználva. Itt is rejtett Markov-modellt alkalmaznak, ami a tudomány sok területén nagyon hasznos, ahol valamilyen adatok időbeli folytonosságát kell leírni, ezen túlmenően még tanuló rendszerek esetén is alkalmas. A szerzők a Markov-láncot itt a „határérték-probléma" kiküszöbölésére is alkalmazzák. Ez a probléma akkor jön elő, amikor valamilyen alkalmazott szegmentáló algoritmus egymást követő képkockák eltérését vizsgálja. Ilyenkor nehéz megtalálni azt a határértéket az eltérésre, amely felett már új *shotról* (jelenethatárról) beszélhetünk. A hangokat nem szemantikus kategóriákba sorolják (beszéd, zene, zaj stb.), hanem vektorokat számítva belőlük egyfajta távolságot definiálnak az egymást követő hangok között hasonlóan a képi feldolgozás során használt képinformáció alapján történő *frame-távolságokhoz.*

Harmadik modalitásként még érdemes megemlíteni a videóban megtalálható szöveges tartalmakat, amelyek szintén felhasználhatók az elemzésben, rendszerezésben. A felismerésük többféle módon történhet. A videóhoz kapcsolt külső adatként elérhető metaadatok vagy különböző nyelvű feliratok nem tartoznak a tartalmi elemzésbe. Viszont a tartalmi elemzéshez tartozik, ha a szöveg a videóállomány szerves részét képezi, azaz byte folyamban van kódolva; ebben az esetben a szövegek felismerése jóval nehezebb, de szerencsére elég sok szöveg- és karakterfelismerő (*OCR – Optical Character Recognition* [Mori et al. 1999]) algoritmus létezik már, jó példa erre az Adobe Acrobat pdf-formátuma által használt módszer. A tartalomban előforduló szövegek többfélék lehetnek:

- Mesterséges szöveg – képeken kívüli (nem a képek szerves részét képező) szövegek, amelyeket tipikusan utófeldolgozás útján „égettek” a filmre. Ezek általában szabványos, egyszínű és statikus karakterek, felismerésük viszonylag egyszerű szabványos betűtípusok esetén. Fajtái:
  - kezdeti: általános, tulajdonképpen „meta” információk, úgymint filmcím, szereplők, videótartalom készítői stb.
  - köztes: általában ahhoz a jelenethez kapcsolható a tartalma, amely felett épp megjelenik. Például helyszín neve, a jelenetben szereplők neve, párbeszédek stb.
  - végén: közreműködők, stábtagok, gyártók, támogatók.
- Jelenetszöveg – a videó szerves része, a kamerával rögzített szöveges információk, amelyek a háttéren és tárgyakon találhatók. Ilyenek például a terméknevek, márkák, helységnévtáblák. Itt sokszor előfordulnak egyedi karakterek is, ráadásul különböző színben, formában és szögekből mutatva, így OCR-módszerekkel igen nehéz a felismerés.

A szöveges információkat ugyancsak jól fel lehet használni. A szöveg mint a három közül legrégebben jelen levő modalitás ma már igen jól kezelhető és rendszerezhető különböző szövegbányászati metódusokkal. Videók esetében általában kisebb a kifejezőereje, mint a másik két modalitásnak, de kiegészítő (ritkán fontos) információkat tartalmazhat, amelyeket érdemes lehet kinyerni és feldolgozni valamilyen módon.

5. Arcfelismerés kameraképeken

Míg az ember számára az arcfelismerés semmilyen nehézséget nem jelent, a számítógépes arcfelismerés sok problémába ütközik. Hogy az arc melyik részét, milyen arcvonásokat kell kódolnunk a számítógépes arcfelismerés során az arcok megfelelő elkülönítéséhez, tudnunk kell, mik azok, amik megkülönböztetnek két arcot. Mi alapján különbözteti meg az ember két társának arcát? Nem csak észleljük az egyes arcrészeket, szemeket, orrot és szájat, hanem az arcvonásokat, illetve térbeli relatív elhelyezkedésüket holisztikusan, egy komplett egészként fogjuk fel. [Parr 2011]

5.1. Arcfelismerés alapfogalmai

Az arcok igen különbözők. Egy személy lehet közeli ismerős vagy teljesen ismeretlen, szomorú, boldog, különböző rasszba tartozhat, lehet fiatal, idős, bajuszos, szemüveges, nézhetjük szemből vagy profilból stb. A folyton változó tulajdonságok ellenére is felismerjük a tömegben a rég látott ismerősünket. Míg az arcon egyszerre jelen lévő számtalan jellemzőnek a gyors elemzése és az arcok, illetve hozzájuk kötődő tulajdonságok felismerése legtöbbünk számára egyszerű folyamat, addig egy arcfelismerő programnak ez korántsem ilyen egyszerű, és az átlagos (azaz csúcstechnológiát nem használó) szoftverek felismerőképessége az emberétől elmarad.

A számítógépes arcfelismerés a gépi látás megjelenésével egyidős. Mind praktikussága, mind pedig a kognitív tudományos érdeklődés erős indíték volt létrejöttében. Bár az emberek az arcukon kívül más jellemzőik alapján is felismerhetők számítógéppel, mint például ujjlenyomat vagy írisz alapján, amelyekkel pontosabb felismerés érhető el, az arcfelismerés mindig fontos kutatási terület marad. Hiszen sokkal természetesebb, kevésbé beavatkozó eljárás, mint az előbb említett biometrikus azonosító módszerek, valamint az emberek számára ez az elsődleges jellemző, amely alapján felismerjük a személyeket. Az iPhone legújabb modelljei (a 2017-ben kiadott iPhone

8 és iPhone X) szintén áttértek az ujjlenyomat-alapú azonosítás helyett a mobiltelefonon futtatható arcalapú azonosításra.

Az első félautomata arcfelismerő rendszert az 1960-as években fejlesztette ki Woody Bledsoe, Helen Chan és Charles Bisson. [Bledsoe 1966] A rendszer működéséhez manuálisan be kellett jelölni a felismerendő arcon bizonyos pontokat; például a pupillák közepét, szemsarkokat. Ezekből 20 távolságértéket számított ki a számítógép 40 kép/óra sebességgel. Ezeket a távolságokat az adatbázisbeli távolságokkal összehasonlítva történt a felismerés. [Bledsoe 1966] Kirby és Sirovich 1987-es munkája mérföldkő az arcfelismerés kutatásában: Karhunen–Loève transzformációt alkalmazva megmutatták, hogy egy teljes arc leírható 100-nál kevesebb értékkel is. [Sirovich–Kirby 1987] Ezt felhasználva Turk és Pentland 1991-ben egy megbízható, közel *real-time* alkalmazást hozott létre az arcok megtalálására, követésére és felismerésére. [Turk–Pentland 1991] Ennek hatására az arcfelismerés kutatása iránt robbanásszerűen megnőtt az érdeklődés. A számítógépes arcfelismerés ipara nagyot lépett előre az elmúlt években, azonban még pontosabb és gyorsabb rendszerekre továbbra is szükség van.

A következőkben az arcdetektálás és arcazonosítás területeihez kapcsolódó módszereket és technikákat mutatjuk be, különös tekintettel a megértésükhöz szükséges alapfogalmakra. Az arcfelismerés témakörébe több egymáshoz kapcsolódó probléma tartozik, amelyek a következők:

- *Arclokalizáció (face localization):* célja a bemeneti tetszőleges képen található arc pozíciójának meghatározása. Csak egyetlen arc helyét találja meg.
- *Arcdetektálás (face detection):* a bemeneti tetszőleges képen szereplő összes arc helyének meghatározását jelenti. Az arclokalizáció továbbfejlesztett változata.
- *Arcazonosítás (face identification):* a bemeneti kép egy arc, amelyet egy adatbázisban szereplő arcképekkel hasonlítunk össze, és egyezést (hasonlóságot) keresünk. A bemeneti archoz a hozzá legjobban hasonlító adatbázisbeli arc címkéjét rendeljük. [Turk–Pentland 1991] [Yang et al. 2002]

Egy általános arcfelismerő rendszer három alrészre bontható: arcdetektálás, jellemzők kinyerése, arcazonosítás. Az első két fázis, az arcdetektálás és a jellemzők kinyerése egyszerre, szimultán is elvégezhető. Az arcazonosító alrendszer pontosságát az arcdetektálás pontossága nagymértékben befolyásolja, hiszen egy nem arc objektumot az arcazonosító mindenféleképpen

helytelenül fogja az egyik személyként felismerni. Így értelemszerűen a teljes rendszer megfelelő működésében mindkét komponensnek igen meghatározó szerepe van.

Az arcfelismerő rendszereket az élet számos területén sikeresen hasznosították, ipari környezetben is, ahogy ezt az alábbi felsorolás példái is mutatják:

- ember-számítógép interfész, videóindexelés, képadatbázisok vizsgálata, [Jafri–Arabnia 2009]
- biztonsági berendezések: létesítmények belépési engedélyének kezelése, repülőterek, határok ellenőrzése, ATM-felügyelet, CCTV-megfigyelőrendszerek,
- személyazonosság igazolása: digitális igazolványok, bankügyletek esetén személyazonosítás, újszülöttek azonosítása,
- specifikus feladatok ellátása az arcfelismerő rendszer kisebb módosításával:
  - nemek szerinti osztályozás, [Moghaddam–Yang 2002]
  - arckifejezés-felismerés [Shinohara–Otsu 2004] (például gyógyászatban [Morik et al. 1999]),
  - arcvonások felismerése és követése [Bourel et al. 2000] (például járművezetők fáradtságának jelzése [Singh–Papanikolopoulos 1997]).

A rendelkezésre álló arcdetektáló és arcazonosító algoritmusok segítségével elkészíthető egy olyan rendszer, amely a térkamerák által adott *streamben* szereplő összes járókelő felbukkanásait és eltűnéseit megjegyzi. A megoldás két oldalról is megközelíthető: ha van információnk a járókelőkről, illetve ha nincs. Előbbi esetére valamilyen arcfelismerő algoritmus használható, ahol a járókelők arca összevethető egy arcadatbázisban szereplő képekkel; utóbbira pedig egy klaszterező alkalmazás szükséges. A járókelők arcainak megtalálására mindkét esetben a megvalósítást arcdetektálással kell kezdeni.

5.2. Arcdetektálás

A Viola–Jones-detektor [Viola–Jones 2001] egy Haar waveleten alapuló kaszkádolt osztályozó AdaBoost tanulóalgoritmussal. Ez a Paul Viola és Michael Jones szerzőpáros által kifejlesztett módszer egy gyors és megbízható gépi tanulási módszer vizuális objektumok felismeréséhez. A módszer alapját a téglalap alakú régiókból képzett jellemzők alkotják. Ezeknek több

típusát is megkülönböztetik, ilyen a legegyszerűbb „két téglalapos”, de fontos szerepet játszanak a három és négy régióból álló jellemzők is. Kinyerésük igen számításigényes feladat lenne, azonban a szerzők a számítás során egyszerre vizsgálják az eredeti képet és az úgynevezett integrált képet, amelyek segítségével ezen jellemzők számítása jelentősen felgyorsítható. A detektor többféle tárgy felismerésére is betanítható, de főként arcdetektálásra használják. A detektor három fő építőeleme a következő:

- integrálkép létrehozása,
- osztályozók tanítása AdaBoost-alapú tanulóalgoritmussal,
- az osztályozók kaszkádstruktúrába építése.

A detektor bemenete a vizsgálni kívánt képkocka, kimenete pedig egy lista a talált arcokat befoglaló dobozokról. Az arcdetektor igen jó hatékonysággal képes megtalálni egy képen az arcokat, így elemzéskor a megtalált arcok és a képkocka egymás közötti viszonyából értékes információkat nyerhetünk ki az elemzéshez, akár a plánok meghatározására is. Hasonló plánelemzést végeztek SVM-alapú klasszifikációval, amely focimeccsek snittjeit képes négy kategóriába sorolni: messzi, közepes, közeli és közönség kategóriákba. A módszer adaptív színfelismerést és hisztogramokat használ [Nan et al. 2008], és mindez egy okos város sportmérkőzéseinek közvetítésénél hasznos alkalmazás lehet.

5.3. Érzelemfelismerés arcokon

A gépi tanuláson belül az érzelemfelismerés egy olyan kutatási területet, amely a számítógépes tudomány, a pszichológia és a kognitív tudomány metszéspontjában áll. Ezen a területen a cél olyan rendszerek és eszközök kifejlesztése, amelyek felismerik, értelmezik, feldolgozzák, illetve szimulálják az emberi érzéseket, és képesek arra, hogy érzelmeket érjenek el.

Az arcokon az emberi érzelem elemzésére, az úgynevezett érzelemfelismerésre az utóbbi 1–2 évben fokozott figyelmet szenteltek a kutatók, mivel számos fontos alkalmazás, mint például az ember és számítógép közötti interakció lehetősége rejlik ebben. A köztereken a közbiztonság miatt ez szintén fontos lehet, hiszen, ha dühös emberek verekedését egy biztonsági kamera venni tudja, akkor akár automatikus rendőrségi riasztás is megoldhatóvá válik ezáltal. A közösségi média tartalmaiban (legyen az akár egy városhoz kapcsolódó tartalom) szintén lehet érzelemfelismerést végezni,

ami gazdasági előnnyel is járhat. Nem kétséges, hogy a képek és videók egyre nagyobb arányban képviselik a médiatartalmakat, így egyre több erőfeszítésre van szükség azok kezeléséhez és a komplexebb feladatok (mint amilyen az érzelemfelismerés) megoldásához.

Korábban is foglalkoztak már a vizuális és multimédiás tartalmak érzelmi felismerésével (arcképen vagy beszédben keresték az emberi alapmintákat), az esetek sokfélesége miatt azonban nem értek el jelentős eredményeket. Napjainkban a mesterséges intelligencia technológiái – például a gépi tanulás – fellendülésének köszönhetően viszont a kutatás e területe ismét fókuszba került. Mivel a számítógép sokkal több megfigyelés elemzését képes elvégezni, pontosabb rendszer alakítható ki az egyes érzelmek felismerésére.

A torontói Ryerson Multimédiás Kutatólaboratórium (Ryerson Multimedia Research Lab) egy érzelmi videó-adatbázist készített és tett publikussá, így egy olyan adathalmaz vált elérhetővé az ezen a területen dolgozó kutatók számára, amelyből egy gépi tanulóalgoritmus képes lehet megtanulni az emberi arcokon észlelhető érzelmeket. Az adatbázis nyolc emberből álló 720 videómintát tartalmaz, akik hat különböző nyelvet beszélnek. A 3–6 másodperces videóklipekben a hat alapvető emberi érzelem fejeződik ki: harag, undor, félelem, boldogság, szomorúság és meglepetés. Az összes videófájl AVI-formátumban található, és összesen 4,2 GB méretű. [Zheng et al. 2014]

Az ilyen jövőbe mutató rendszereknek több alkalmazási lehetősége is nyílik egy okos városban. Hasznos lehet az okos város egyik kulcsterületénél, a közlekedésnél. A járművezetői megfigyelő rendszerek ma már léteznek, de általában csak akkor avatkoznak közbe, ha a járművezető zavart vagy fáradt. Az érzelemfelismerő rendszert az autóba lehet telepíteni (úgy, hogy a kamera a vezető felé nézzen), és a rendszer meghatározhatja, hogy a vezető dühös, szomorú, boldog, meglepett vagy érzelemmentes. A balesetek nagy része a vezető figyelmetlensége vagy nem megfelelő érzelmi állapota (például düh) miatt következik be, ezért a járművezető érzelmeire vonatkozó információk felhasználhatók lennének az autó ütközés előtti műveleteire, például a biztonsági övek meghúzására vagy a fékezésre. A jövő autonóm autója akár át is veheti a vezetést, ha úgy érzékeli, hogy az ember nem felel meg a feladatnak. Másik alkalmazási lehetőség az okos mobiltelefonok területén jelenhet meg. Ha a minket körülvevő okos tárgyak képesek az érzelmeinket felismerni, akkor egy adaptív viselkedés beépítésével (beprogramozásával) testre szabhatóvá válhatnak, akár egyénre szabottan, akár időben változóan úgy, hogy a készülék máshogy viselkedne, ha a gazdája szomorú, vagy éppen vidám.

Vákát

6. Videótartalmak elemzése, áttekintése

Az okos város polgárai által készített *homemade* videók rendszerezéséhez, gyors áttekintéséhez számítógépes támogatás szükséges. A ma használatos számítógépes operációs rendszerekre jellemző, hogy valamennyire már segítik a videók gyors áttekintését, tömörítését, kivonatolását, a lényeg kiemelését *(video skimming).* Ehhez egy előnézeti képet generálnak, ami igen gyakran csak az első képkockát jelenti, amely általában fekete és sok információt nem hordoz. Előfordul még a véletlenül kiválasztott képkocka is. A gyors áttekintésre egy másik megoldás, hogy a videókból az eredeti hosszuknál jóval rövidebb, úgynevezett összegző jelenetsort készítünk. Ennek segítségével megérthetjük, sokkal könnyebben áttekinthetjük akár a saját magunk által készített családi videókat, akár a mások által nyújtott klipeket és egyéb típusú videókat. Ezen módszernek a lényege, hogy egy kép helyett mozgóképrészleteket emelünk ki az eredetiből.

Az előnézeti kép vagy videó készítése kézzel igen hosszadalmas feladat lenne, ezért érdemes egy automatikus, gép által elvégezhető módszert készíteni. Ezzel annyi a probléma, hogy a gép sokkal nehezebben képes értelmezni a látott videót, a tartalmi összefüggések kinyerése még ennél is nehezebb feladat. További nehezítés, ha a videóról semmiféle metainformáció nem áll rendelkezésre, mert ilyenkor pusztán a kép vizuális információi alapján kell következtetéseket levonni. Mivel a feladat meglehetősen szubjektív (mi számít egy jó összegzésnek), a módszer minőségi jellemzése igen nehéz.

Ebben a fejezetben a megoldások megértéséhez szükséges módszerekről és technikákról adunk egy rövid bemutatást kezdve a videószegmentáló algoritmusok és módszerek ismertetésével, majd a jellemző információk kinyerését célzó legelterjedtebb képfeldolgozási technikákkal folytatjuk, végül az összegző jelenetsor generálásával fejezzük be.

6.1. Videószegmentálás

A videótartalmak elemzéséhez, az emberi megértés támogatásához, az információ kinyeréséhez a teljes videót szegmentálni, részekre bontani szükséges.

A legjellemzőbb, legalapvetőbb módja ennek a felvételek detektálása. Egy felvétel alatt azt a részt értjük, amely a kamera elindulása és megállása között van. A felvétel határa általában egy vágás, snitt vagy valamilyen átmenet, átúsztatás (azonban az egész felvételre is használatos a snitt szó, esetleg az angol *shot* kifejezés). Egy felvételen belül a kamera, és a képen szereplő objektumok is mozoghatnak, illetve új objektumok is feltűnhetnek.

A videótartalmak egy másik lehetséges felbontása a jelenet, amely egy magasabb szintű felbontás, és inkább szemantikai megközelítése a problémának. Alapvetően itt is először felvételekre kell felbontani a videót, majd ezeket kell csoportokba, azaz jelenetekbe rendezni. Egy jelenet például tartalmazhat egy párbeszédet, amely ugyanazon helyszínen azonos időben történik különböző kameraállásokból, snittekből. Mind a felvétel, mind a jelenetek detektálása kézi módszerrel túlságosan időigényes, nem praktikus, ezért van szükség a gépi, automatikus szegmentációra. E kettő közül az utóbbi, azaz a jelenetek detektálásának automatizálása a nehezebb, így ebben a tanulmányban csak a felvételszintű szegmentálással foglalkozunk. Azon belül is a küszöbértéket használó szegmentáló módszereket ismertetjük.

A küszöbértéket használó módszerek lényege, hogy egy-egy mértéket definiálnak két képkocka között, és ha ez az érték átlép egy meghatározott küszöböt, akkor állapíthatunk meg felvételhatárt. Ezen küszöbértékek megállapítása nagyon függ a videótartalomtól, így általánosan nehéz jó értéket találni, azonban a fix számokat használó módszerek mellett léteznek úgynevezett adaptív küszöbértéket használó módszerek is. [Natarajan 2009] A [Dailianas et al. 1995] cikk többféle paraméter szerint, részletesen összehasonlítja ezen technikákat, amelyek a következők:

– *Intenzitáskülönbség-alapú szegmentálás.* A felvételhatárok detektálásához az egyik legegyszerűbb módszer a szomszédos képkockák pixelei intenzitásának abszolút különbségeit vizsgálja. Ha ez a különbség nagyobb egy meghatározott értéknél két képkocka között, akkor abban a pozícióban nagy valószínűséggel vágás található.
– *Hisztogramalapú szegmentálás.* Szintén egyszerű és hatásos módszerek a hisztogramalapú módszerek. Az alapötlet a következő: minden képkockához konstruálható egy hisztogram, amely az adott képkocka konkrét színértékeinek az előfordulásait számolja. Ezeket a hisztogramokat minden képkockára kiszámítjuk, majd az egymás után következőket összehasonlítjuk. Ezen eloszlásértékek egy-egy felvételen belül igen hasonlók lesznek még abban az esetben is, amikor a kamera vagy a képen lévő objektum mozog, mivel ez a módszer érzéketlen

az objektumok képen belüli pozíciójára. Ezután már csak össze kell hasonlítani két egymást követő kép hisztogramját, amire több variációs módszer is létezik, de legegyszerűbb módszer a hisztogramok abszolút különbségének kiszámítása. Ha a különbségek összege egy küszöbértéknél nagyobb, akkor ott szegmentációt hajtunk végre.

– *Momentumalapú módszerek.* A valószínűség-számításbeli momentumszámítási módszerek alkalmazásáról van szó a számítógépes látás témakörében. A képmomentum kifejezés ezen formulák képekre való alkalmazását jelenti. Egy ezzel foglalkozó publikációban [Arman et al. 1994] ezekkel a képmomentumokkal kapcsolatos szegmentálási módszert mutatnak be. A momentumok rendelkeznek azzal az igen hasznos tulajdonsággal, hogy függetlenek a skálaváltozásra, a forgatásra és az eltolásra. A momentumokból vektort képezhetünk, és két szomszédos kép távolsága ezen két vektor közötti euklideszi távolság lesz.
– *Éldetekció-alapú szegmentáció.* Az elterjedt módszerek körébe tartoznak még az éldetekció-alapú módszerek. Ezeknek a lényege az, hogy minden képkockára megkeresik a domináns éleket, majd a szomszédosaknál összehasonlítják azokat. [Mai et al. 1995] Az összehasonlítás alapján elég jól kitalálható, hogy a két egymást követő kép azonos felvételbe tartozik-e, vagy köztük egy vágás van, például mert a felvételek határainál az eddigi élektől távol újak jelenhetnek meg a képen, illetve régi élek tűnhetnek el.
– *Képszeletalapú módszer.* A [Lu et al. 2004] dokumentumban a szerző egy másik, jóval egyszerűbb szemszögből közelítette meg a problémát. Az elmélete szerint elegendő a képből egy szeletet kivágni, például vízszintesen a közepén. Minden képkockát ugyanott vág el, és csak ezen szeleteket hasonlítja össze. A képkockák közötti távolság mértékét a szeletek pixelkülönbsége alapján definiálja a szerző, így egy sokkal gyorsabb megoldáshoz jut, bár az eredmény nem lesz annyira pontos.

6.2. Összegző jelenetsor készítése

Egy videó gyors áttekintéséhez érdemes az eredetinél rövidebb, kivonatolt, összegző jelenetsort készíteni. Erre a problémára már születtek megoldások, ötletek a szakirodalomban, amelyeket ebben az alfejezetben ismertetünk.

6.2.1. Videókivonatoló módszerek

Először bemutatjuk a primitív videókivonatoló, összegző *(video skimming/ summarization)* módszereket, ezután a prioritásalapú módszert és végül a hangalapú módszereket. A felügyelet nélküli és a felügyelt gépi tanulásos eljárásokat a következő két szakaszban ismertetjük.

Primitív videóösszesítő módszerek:

- A videók gyorsabb áttekintéséhez az egyik legegyszerűbb eljárás a kép és hang felgyorsítása. Ezzel a módszerrel csak kismértékű, kb. 1,5–2,5-szeres gyorsítás érhető el, hogy követhető legyen. Ez kevés, valamint nem is praktikus, így ennek a módszernek egy kisebb továbbfejlesztéseként kihagyják azokat a jeleneteket, amikor csend van, vagy nem történik semmi a videóban.
- Egy másik szintén egyszerű módszer, amikor a videótartalomból csak képeket vágnak ki, és ezeket illesztik egymás után bizonyos ideig egy diavetítést létrehozva. E megoldás jelentős hátránya a felhasználók szemében, hogy a videó dinamikája jelentősen csökken, sok információ elveszik.

Prioritásalapú módszer: Az [Albanese et al. 2006] cikkben a szerzők egy olyan módszert fejlesztettek ki, amely úgynevezett prioritásgörbe alapján képes összegezni bármilyen videót. A módszer több lépésből áll. Az első lépés, hasonlóan a korábban ismertetett módszerekhez, a blokkokra bontás. Ez lehet akár egyenlő hosszúságú részek kivágása, vagy akár egy szegmentáló eljárás kimenete, a lényeg, hogy viszonylag rövid részeket kapjunk.

Ezután második lépésként a blokkokhoz prioritásértékeket kell rendelni. Ez történhet automatikusan a tartalom alapján, például témafelismerés vagy metainformációk alapján, de történhet kézzel is, emberek által. Egy jó példa erre a labdarúgás-közvetítések esete, ahol a prioritás meghatározás lehet a következő: gól: 10, piros lap: 7, sárga lap: 6, szöglet: 3, verekedés: 10. Ebben a példában a hozzárendelés kézzel egyértelmű, ám időigényes, azonban ezen események detektálása automatikusan is megoldható néhány alfeladatnál, mivel a kameraállások általában adottak, a pálya szinte egyszínű zöld, így könnyen szegmentálható, továbbá valamelyik sarokban mindig ott van az eredmény is, amelynek a változását felismerve egyértelmű, hogy gól történt.

A következő, harmadik lépés a csúcsok keresése *(peak detection)*, ahol a blokkokhoz hozzárendelt prioritások alapján keresünk kiugró szegmenseket. Egy szegmens prioritását a bennük levő blokkok prioritásának átlagaként számoljuk, majd egy választott küszöbszám alapján minden olyan szegmenset eldobunk, amely kisebb prioritással rendelkezik, mint ez a küszöbszám, és a maradékok lesznek a kiválasztott szegmensek.

A negyedik lépés: általában adott egy fixen meghatározott vagy akár az eredeti videó hosszával arányos hosszúság, amit be kell tartani az összegző jelenetsor generálása során. Ehhez a prioritásokkal arányosan le kell kicsinyíteni a szegmensek méretét (levágni az elejéből és a végéből), hogy a végleges montázs beleférjen az időkeretbe.

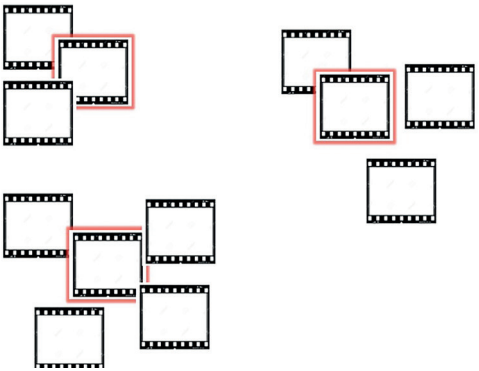
Hangalapú módszerek. A szakirodalomban olvashatunk olyan videóösszegző módszerekről is, amelyek a videó hangját is felhasználják. [Smith et al. 1998] [Taskiran et al. 2002] Ehhez általában a hangsávban lévő szüneteket, illetve az emberi beszéd felismerését használják fel.

6.2.2. Kiválasztás felügyelet nélküli gépi tanulással (klaszterezéssel)

Az összegző jelenetsor generálásának kulcslépése, hogy a felvételek közül kiválasszuk, hogy melyik legyen benne a végeredményben, és melyik ne. Fontos szempont, hogy az összegzés a lényeget próbálja meg összefoglalni, ismétlődő részeket lehetőleg ne tartalmazzon. Erre a problémára egy lehetséges jó megoldás, ha a kinyert videószegmenseket hasonlóság alapján csoportokba rendezzük, majd ezen csoportokból választjuk ki az elemeket. Ennek megvalósítása a klaszterezés módszerével történhet, amelyre a legismertebb algoritmus a k-átlag *(k-means).*

A k-átlag algoritmusnak paraméterként meg kell adni egy kiindulási klaszterezést mint bemenő paraméter, valamint azt, hogy hány klaszterbe rendezze az elemeket. Az előbbi viszonylag egyszerűen megtehető egy véletlenszerű csoportosítás megadásával. Az utóbbi azonban általában nem könnyű feladat, ezért erre ökölszabályként széles körben alkalmazzák azt, hogy legyen a *k* klaszterszám a pontok (jelen esetben a felvételek) számának négyzetgyöke, illetve másik megoldásként a pontok száma felének a négyzetgyöke. Ezen kívül a klaszterezéshez szükséges még a pontok közötti távolságok számítási módjának megadása, azaz definiálni kell egy távolságfüggvényt, amely a legtöbb esetben euklideszi távolság szokott lenni.

A klaszterezés sikeres végrehajtása után a felvételeket a klaszterező módszer hasonlóság alapján csoportokba rendezi, azonban az a feladat, hogy melyik felvételt válasszuk be az összegző jelenetsorba, még hátravan. Erre a legegyszerűbb megoldás, hogy minden klaszterből egy elemet választunk ki, méghozzá a klaszterek centroidjához (az elemek összességének aritmetikai középpontjához) legközelebbi elemet célszerű kiválasztani, ahogy az 5. ábrán is látható (minden klaszternél a piros kerettel megjelölt kép van a legközelebb a centroidhoz, így ezek lesznek kiválasztva). Valós alkalmazások során előfordulhatnak kis elemszámú klaszterek is, amelyek az esetleges, ritkán előforduló egyedi eseteket jellemzik. Így nem érdemes az összes klaszterből választani, inkább csak néhányat érdemes felhasználni. A klaszterek közül azokat célszerű beválasztani, ahol az elemek száma nagyobb egy bizonyos küszöbértéknél; erre az összes elem 10%-a körüli értéket érdemes megadni.



5. ábra

Klaszterezett képek (piros kerettel megjelölve a centroidhoz eső legközelebbi képeket)

Forrás: a szerzők szerkesztése

Klaszterezésalapú összesítést alkalmaztak két másik megoldásnál is [Beran et al. 2007] [Chheng 2007], ahol az összegző jelenetsor készítésénél a jelenetek hasonlóságát vették alapul. A klaszterezés előkészítéséhez először a képkockákból jellemzőket nyertek ki (általában a színek hisztogramját és a gradienseloszlást szokták erre a célra használni), és ezek összefűzéséből egy „képszintű leírót” definiáltak minden képkockához. Ezután bizonyos képeket eldobtak, ami úgy történik, hogy előre definiálnak „nemkívánatos képkockákat” (például egyszínű kockák, csíkos képkockák), majd

összehasonlítják őket minden képkockával előre megadott jellemzők alapján. Ha a hasonlóság egy bizonyos értéknél nagyobb, akkor az a képkocka „nemkívánatos". A felvételhatárok kockái is a „nemkívánatosak" közé tartoznak. A jellemzőkinyerés után a videót rövid részekre bontják, ezeket tekintve egy-egy felvételnek. A nemkívánatos képkockák kiszűrése után a maradékra egy-egy „felvételszintű leírót" definiálnak. Ez a magasabb szintű leíró a felvétel által tartalmazott képkockák leíróinak az összefűzéséből adódik. A felvételekre k-átlag klaszterező eljárást alkalmaznak euklideszi távolsággal. A klaszterek számának megadását a videó hosszából, illetve a felvételek számából tehetjük meg a már említett négyzetgyökös ökölszabály szerint. A klaszterezés végén a kialakult csoportokból a legjellemzőbb jelenetet (azaz amelyik a legközelebb volt a centroidhoz) választották ki (hasonló módon, mint a képhalmazokból kiválasztott reprezentáns képeknél [Szűcs et al. 2011], és tették be őket az összesített videóba. Ezen felvételek sorrendjét az eredeti videóban elfoglalt pozíciójuk alapján adták meg.

6.2.3. Kiválasztás felügyelt gépi tanulással (osztályozással)

Az összegző jelenetsorba való beválasztást meg lehet oldani felügyelt gépi tanulás segítségével is, ahol a bemenet a felvételek (szegmensek), a kimenet pedig egy bináris információ minden felvételhez, hogy az adott szegmens az összegző jelenetsorba való-e, vagy sem. A tanításhoz szükséges néhány meghatározott tulajdonság kiválasztására. Ezekkel lehet majd az osztályozót betanítani, és ezen attribútumok alapján határozza meg egy felvétel osztályát is. Az egyes felvételeket jellemző attribútumok:

- relatív pozíció: a felvételnek a relatív kezdési pozíciója, értéke 0 és 1 közötti lehet;
- hossz: a felvétel hossza képkockákban számolva;
- arcok száma: a felvételen megjelenő arcok száma;
- dinamika: a felvétel sebessége, illetve változékonysága (ez a többi attribútumhoz képest kiugróan más értékekkel rendelkezik, ezért normalizálni kell a felhasználás előtt);
- teljes hossz (például másodpercben).

6.2.4. Végső összefűzés

Az összegző jelenetsor generálásához a felvételek kiválasztása után az utolsó lépés azok összeillesztése egy új összefüggő videófájlként. A klaszterező vagy az osztályozó kimeneteként kiválasztott felvételek fognak bekerülni az összegző jelenetsorba. Mivel az összegzésben is sorrendhelyesen célszerű megjeleníteni az egyes részleteket, mindenképpen rendezni kell a megjelölt felvételeket az eredeti videóban elfoglalt helyük szerint.

Első lépésként végigmegyünk mindegyik kiválasztott szegmensen úgy, hogy a korábban letárolt kezdési idő és hossz alapján kivágjuk a kérdéses részt az eredeti videóból, egy új fájlt létrehozva. A kivágáshoz és a többi hasonló művelethez az *ffmpeg* nevű alkalmazást [Ffmpeg] érdemes használni, amely egy széles körben alkalmazott program hasonló feladatokra, és több médialejátszó szoftver is ezt használja a háttérben. Jól paraméterezhető, sok funkciót megvalósító parancssoros felülettel rendelkezik, így az alkalmazásokba könnyen és egyszerűen integrálható.

A különböző formátumú videófájlok nehezen összeilleszthetők, ezért minden kivágott szegmenst egy közös formátumra (például MPEG-formátumba) érdemes átkonvertálni. Miután az összes kiválasztott felvételből létrehoztuk a köztes fájlokat, szintén az ffmpeg program segítségével ezek könnyen összekonkatenálhatók ismét egy fájlba. Utolsó lépésként az összefűzött fájl újra átkódolható bármilyen formátumba (természetesen a létrehozott ideiglenes videófájlok törölhetők), amire szüksége van a felhasználónak.

A generált összegző jelenetsor éles vágásokat tartalmazhat, gyakran teljesen különböző jeleneteket is összefűzhet. Ha speciálisan a filmek összegző jelenetsorát, úgynevezett trailert kell automatikusan előállítani, akkor a megtekintés során zavarók lehetnek ezek az éles vágások, beszéd esetében különösen. Ennek javításául szolgál, ha ezen részek között mesterséges átúsztatást illesztenek be, például átfedéses átúsztatást. A mozifilmekben gyakran előforduló eset az, amikor még a vágás előtt elindul a következő jelenet hangja. Az ilyen fajta hangátúsztatással megoldható lenne a probléma, mivel az átfedéses képátúsztatás során lehetséges előbb elkezdeni a kérdéses jelenetet már a hanggal együtt.

Az összegző jelenetsor generálása számos videótartalommal kapcsolatos területen felhasználható. Erre példa az online videómegosztó oldalak, ahol az esetlegesen hosszabb műveket gyorsan áttekinthetjük az összegzés által. Hasonló példa a digitális műsorszolgáltatás (azon belül a VOD – *Video on Demand*), ahol előfordulhatnak előzetessel nem rendelkező videók, például

sportközvetítések, szórakoztató show-műsorok, vetélkedők. A videókban való keresés is igen fontos funkció a videómegosztó portálokon, bár képalapú keresésre kevés példát találunk. A következő fejezetben mutatjuk be a keresés, többek között a kép alapján történő keresés menetét és módszereit.

Vákát

7. Keresés videótartalmakban

Videótartalmakban alapvetően kétféle keresésről beszélhetünk, egyrészt a képen látható objektumok, tárgyak, személyek felismerésén alapuló keresésről, másrészt adott pillanatképek megkereséséről a videóban.

7.1. Tartalomalapú képvisszakeresés

A szöveges tartalmak elemzésére már jó ideje készítenek különböző cégek kereső és indexelő szolgáltatásokat, az egyik legjobb példa erre az internetes keresők. Azonban a képek és videók kezeléséhez más módszerek szükségesek. Ezen tartalmakról információt kinyerni/kereshetővé tenni teljesen más technikákat és megoldásokat igényel, mint szövegek esetén. A sokkal összetettebb és nagyobb adathalmazok miatt természetesen ezek nehezebb és nagyobb számítási igényű feladatok, ahol új problémák is felmerülnek. Míg szövegek esetén főként szavakat, mondatokat nyerhetünk ki szövegbányászati módszerek segítségével, addig képeknél a kép tartalmára vagyunk kíváncsiak. A rendszerrel szemben itt az az elvárás, hogy a képen látható objektumokat, részeket felismerje és valamilyen módon kereshetővé tegye. A témakörben már léteznek különböző megoldások, amelyeket a következőkben mutatunk be.

A képekből való tartalomkinyerés témakörét gyűjtőnéven CBIR- (*Content Based Image Retrieval* – tartalomalapú képvisszakeresés) technikák néven emlegetjük. Ez egy rendkívül szerteágazó és sok módszert magába foglaló témakör. A lényege, hogy a képről információkat gyűjtsön, és azokat valóban a kép tartalmából szerezze meg. A régebbi módszerek, sőt még manapság is a legtöbb webes képkereső szolgáltatás csak a fellelhető képekhez tartozó szöveges adatok alapján tud következtetni a kép információtartalmára: például a képhez csatolt leíró címkék, a képet körülvevő HTML-információk (metatagek, alt attribútumok stb.) alapján. Ezek azonban megbízhatatlanok és pontatlanok lehetnek, vagy egyáltalán nem is állnak rendelkezésre (mint a térfigyelő kamerák képeinél), így a keresés általában nem nyújt kielégítő eredményt.

A CBIR segítségével azonban a kép valós tartalmát tudjuk elemezni (színek, formák, alakok stb.), sőt haladó technikákkal akár konkrét objektumokat, tárgyakat is felismerhetünk. E módszerekkel sokkal több információhoz juthatunk a képek tartalmáról, a kereséseket is sokkal hatékonyabbá tehetjük, és intelligens alkalmazások készíthetők. Idetartozó funkciók a képfelismerés és képkeresés, videókövetés, mozgások felismerése. Alkalmazások sora készült, illetve készül különböző módszerek segítségével.

A kamerák által felvett videótartalmakban való keresés nagymértékben hasonlít a tartalomalapú képkeresési feladatokhoz, mivel a videó tekinthető képek halmazának. Azonban a mozgóképeknél felhasználhatunk további információkat, például a képkockák egymáshoz való időbeli viszonyát.

A képvisszakereső rendszer [Squire et al. 2000] megvalósításának szerzői arra fektették a hangsúlyt munkájuk során, hogy érdemes minél több eredményt és módszert felhasználni a már több évtizede kutatott szövegvisszakeresési témakörből, mivel a probléma részei könnyen lefordíthatók szövegkeresési problémákra. A képekből jellemzőket nyertek ki, például szín, textúra és alakzatinformációk alapján, majd ezek felhasználásával számoltak hasonlóságot a képek között, a szövegkeresési fogalmakat pedig az aktuális problémakör elemeire alkalmazták.

Egy másik megoldás során egy komplex lekérdezőt [Flickner et al. 1995] hoztak létre videókban való kereséshez, amely segítségével képesek vizuális információk, akár kép alapján is keresni. A videók elemzése során fontos lépésként említik a videó szegmentálását – felvételekre bontását – és ezekből a jellemző képkocka kinyerését.

A Video Google-nak elnevezett megoldásnál [Sivic–Zisserman 2006] egy olyan rendszert valósítottak meg, amely skálainvariáns képleírók (SIFT) segítségével képes egy megadott kép összes előfordulását megkeresni egy adott videón belül. A képleírókat vizuális szóként indexelik, szintén a szöveges keresési módszereket újrafelhasználva erre a problémakörre. Ilyen például a stoplista, a dokumentumfrekvencia és az invertált indexállomány.

7.2. Képalapú keresés videótartalmakban

A videókban való keresési feladatok között külön altípus a kép alapján történő keresés, amikor egy meglevő kép előfordulásait kell előkeresni és megjeleníteni a felhasználó számára. A kép megadása egy felhasználói bevitel, amelyet feldolgozva a rendszer válaszol a találatok megjelenítésével.

A képkockák összehasonlítása történhet a 1.1. alfejezetben már bemutatott skálainvariáns képleíró eljárások (SIFT – *Scale Invariant Feature Transform* vagy SURF – *Speeded Up Robust Features*) alapján, amelyek a kép legreprezentatívabb pontjainál, az úgynevezett kulcspontoknál lesznek kiszámítva. Ezeket a leírókat tisztán a kép tartalmi információi alapján (pixelszinten) számolják ki egy magasabb szintű jellemzést adva az adott képről. Ezen jellemzés másik előnye, hogy a különböző képeken megjelenő ugyanazon objektumnál más méretben, más megvilágításban, akár elforgatva nagyon hasonló leírók jönnek ki a számítás során, így ha az objektum két előfordulása vizuálisan nem is egyezik meg teljesen, mégis meghatározhatunk egy mértéket a hasonlóságukra. Ezen mértékekre alapoz a keresési eljárás, és lényegében kizárólag az ezen leírók által hordozott információkat használja fel közvetlenül. A felhasználói bevitel során megadott kép kereséséhez először szükséges a teljes videó előfeldolgozása, mivel a leírók kinyerése az egész videóra igen hosszú ideig is eltarthat.

Ahogy említettük, a SIFT vagy SURF leíró vektorok kiválóan alkalmasak képek összehasonlítására. A videó képkockáira kell lefuttatni az algoritmust, azonban felesleges minden egyes képkockára ezt megtenni: jó eredményt kaphatunk akkor is, ha csak másodpercenként egy képet használunk. A képkockák az algoritmus futtatása előtt szürkeárnyalatosra konvertálhatók, mivel az algoritmus pusztán a pixelintenzitásokkal dolgozik, a színinformációkat figyelmen kívül hagyja. Ezen kívül hisztogramkiegyenlítés is végezhető előfeldolgozási lépésként.

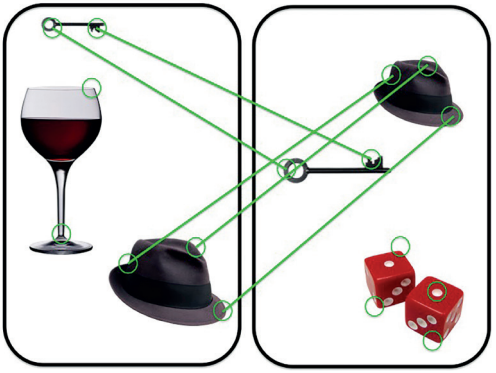
A megkeresendő referenciakép kulcspontjainak kinyerése után azok összehasonlítása a videóból kiragadott minden egyes pillanatkép kulcspontjaival nagyon nagy számításigényű feladat, mivel rengeteg összehasonlítást igényel, aminek a végrehajtási ideje hosszúra nőhet. A rendszer használhatósága érdekében ezt az időt mindenképpen le kell csökkenteni, hogy a felhasználó a keresés indítása után pár másodpercen belül már eredményt kapjon.

7.2.1. Invertált index

A kulcspontok tárolásához invertált indexet érdemes felépíteni a videó képkockáiról és az azokon előforduló leírójellemzőkről. Ennek segítségével egy lépésben megmondható, hogy egy keresett kulcspont mely képkockákon szerepel. Ez egy elengedhetetlen lépése a keresési lekérdezésnek, így ezt

az indexstruktúrát érdemes megfelelő módon letárolni, mivel ennek az előállítása hozzátartozik az előfeldolgozáshoz.

A referenciakép és a találati képkocka (vagy képkockák) kulcspontjai között kell a hasonlókat megkeresni. A kulcspontok leírói (például SIFT-leírók) alapján a legközelebbi szomszédok keresésének (NNS – *Nearest Neighbour Search*) módszerével érdemes ezeket összehasonlítani. [Yianilos 1993]. Miután a leírók összehasonlítása megtörtént, megkaptuk, hogy melyek egyeznek, és melyek nem, ahogy ezt a 6. ábra is mutatja (az egyezőket a jobb láthatóság kedvéért összekötöttük). Két kép közötti egyezés mértékét az egyezések darabszáma alapján lehet megkapni. Ezt inkább relatívan érdemes számolni az abszolút darabszám helyett. Tehát az egyezések arányszámát szokás a hasonlóság mértékének tekinteni. Az arány számítására több jó megoldás is létezik: lehet az összes lehetséges párhoz viszonyítani, vagy lehet a két kép közül a több leírót tartalmazót venni a normalizálás alapjának. A hasonlóság alapján már csökkenő sorrendbe helyezhetők a találatok.



6. ábra

Két kép közötti leírók összehasonlítása

Forrás: a szerzők szerkesztése

A keresés után megjelenített találati listán az egyes képkockák az egyezések száma alapján jellemezhetők, sorrendbe állíthatók, sőt ha a hasonlósági mérték túl alacsony, akkor a küszöbérték alatti találatokat akár el is lehet hagyni. E küszöbérték megfelelő megválasztása külön vizsgálatot igényel.

Az eredmények minőségére két jellemzőt szokás használni a keresők értékelésénél. Az egyik ilyen a *fedés,* azaz hogy a releváns előfordulások

hány százalékát adja vissza a rendszer. A másik jellemző, amit a minőség mérésére használnak, a *pontosság,* amely azt mondja meg, hogy az összes találat közül hány százalék a releváns. A találati lista tartalmazhat olyan képkockákat is, amelyek egyáltalán nem tartalmazzák a keresett objektumot, még csak hasonlót sem, azaz ezek a nem releváns találatok. A fedés és pontosság meghatározásához szükségünk van egy-egy képrészlet valódi előfordulásainak ismeretére, azaz referenciaadatokra. Ezeket természetesen mindenképpen kézzel kell előállítani, mivel fontos, hogy teljes mértékben a valóságot tükrözzék, ugyanis csak így lehet hibátlan mérést végezni.

7.2.2. A képinformációk tárolása

A korábban bemutatott képfeldolgozási megoldások több különböző számításigényű folyamatból állnak, amelyek nagy méretű adathalmazt állítanak elő. Az egyes lépések során előállított adatokat a későbbi alfolyamatok is felhasználhatják, így elengedhetetlen ezen információk megfelelő tárolása. Mivel a tárolandó adathalmaz mérete több gigabájt is lehet, nem célszerű a memóriában tárolni: kizárólag a merevlemezre való rögzítés jöhet szóba. Ezen megoldás mellett szóló érv az is, hogy így az egyes alfolyamatok a későbbiek során újra lefuttathatók az azokat megelőző lépések ismételt elvégzése nélkül. A következőkben röviden bemutatunk néhány jó példát a képinformációk tárolásának megoldásához.

CSV-formátum: a CSV (*Comma-Separated Values* – vesszővel elválasztott értékek) formátum az egyik legelterjedtebb export/import adatformátum táblázatok és adatbázisok mentéséhez. Sajnos nem létezik szabvány hozzá, így a konkrét formátumot a kezelni képes alkalmazások definiálják. Ennek következménye az alkalmazások közötti inkompatibilitás. A leggyakrabban előforduló különbségek például az elválasztó karakterek vagy az idézőjelezés, azonban ezen problémák könnyen áthidalhatók egy programozó számára. Ezen hiányosságok ellenére mégis elterjedt formátummá vált, mivel még így is könnyen és egyszerűen lehet vele táblázat formájú adatokat írni és olvasni.

Adatbázis-alapú tárolás: az adatbázis-alapú tárolás lehetőséget biztosít a több különböző adattípusok rögzítésére, ráadásul a közöttük lévő kapcsolatok megfelelő ábrázolása is megoldott. Léteznek kifejezetten beágyazott körülményekhez kifejlesztett adatbázis-kezelő rendszerek, ilyen például az *sqlite3.* Ez egy C nyelven írt, nyílt forráskódú, magas szinten optimalizált SQL-alapú adatbázisszerver, amely hálózati kommunikációval

nem foglalkozik, konfigurációt nem igényel, kizárólag lokálisan érhető el. Például a SURF-algoritmus által kinyert pontok tárolásánál a leíró vektorok mind 128 dimenziósak, így egy legalább 128 oszlop szélességű táblázatra van szükség. Azonban itt problémát okoz az oszlopok nagy száma, mert az adatbázis-kezelő rendszerek nem ilyen típusú adatok tárolására valók, inkább az egyszerűbb adattípusokra és az azok között lévő kapcsolatokra koncentrálnak.

HDF adattárolási formátum: a HDF5 (*Hierarchical Data Format* – hierachikus adatformátum) egy olyan egyedi technológia, amely lehetővé teszi az extrém méretű és komplexitású adatgyűjtemények kezelését. Egy flexibilis adatmodellből áll a központi része, amely képes nagy bonyolultságú összetett adatok és széles skálán mozgó metaadatok reprezentálására. Definiál egy teljesen hordozható fájlformátumot a benne tárolt adatok mennyiségére és méretére való korlátok nélkül. Széles körben alkalmazzák különböző platformokon az egyszerű laptopoktól egészen a teljesen párhuzamos működésű szuperszámítógépekig. Léteznek magas szintű interfészek és eszközök az adatok elemzésére, megjelenítésére és manipulálására. A HDF5-fájl egy rendezett hierarchikus struktúrát követ, a két legfontosabb eleme a *group* és a *dataset:*

- *HDF5 group:* egy csoportot valósít meg, amely tartalmazhat tetszőleges számú *datasetet*, illetve metaadatokat.
- *HDF5 dataset:* többdimenziós tömb adatelemek tárolására, metaadat-információkkal. Két részből áll, egy fejlécből és magából az adattömbből.

A képkockánkénti elemzés során kinyert abszolút különbségeket az aktuális pozícióval együtt egy HDF5-formátumú fájlba érdemes menteni. Ez igen hasznos, mivel nem kell minden alkalommal a videófájt elemezni, továbbá a végfelhasználás során is hasznos lehet, hogy ha utólag kell újraalkalmazni a többi lépést, például a felvételhatárok detektálását esetleg más paraméterekkel. A szegmensekre való felbontás után a szegmenshatárokat is el lehet menteni ugyanabba a HDF5-fájlba.

8. Kihívások és trendek a multimédiában

A multimédia-tartalmak feldolgozásához kapcsolódó kutatásokat áttekintve a jövőre vonatkozóan nagyon sok lehetőség rejlik a témakörben, sőt alkalmazásaira mind szélesebb területen számíthatunk, amelyben az okos város témaköre kiemelt szerepet kaphat. Léteznek élesben is működő alkalmazások, elég csak a népszerű webes szolgáltatásokra gondolni. Ezek főként a téma egyszerűbb megközelítései, limitált funkcionalitással. Rengeteg alapkutatás is folyik jelenleg világszerte, amelyek ötleteit, eredményeit még nem ültették át a gyakorlatba.

Ebben a fejezetben előbb összegezzük a tanulmánynak a kameraképek feldolgozására vonatkozó eredményeit, majd bemutatjuk a multimédia-technológiák területén felmerülő kutatási kérdésköröket, nemzetközi trendeket, elsődlegesen az okos város témaköréhez kapcsolódóan.

8.1. Az okos város kameraképeinek feldolgozása

Ahogy láttuk, az okos város kameraképeinek elemzéséhez a legfontosabb informatikai terület a kép- és videófeldolgozás. Az előző fejezetekben ennek technológiáit rendszereztük és ismertettük, amelyekhez néhány kiegészítő megjegyzést kívánunk tenni.

Az egyik legfontosabb befolyásolási tényező a képfeldolgozás szempontjából a megvilágítás. Ha a megvilágítási tényezők megfelelők (például a kamera felől érkezik a fényforrás), akkor a képfeldolgozó programok is jól működnek. A megvilágításnál fontos a homogenitás, azaz a lényeg az árnyékok és a becsillanások minimalizálása. A rossz fényviszonyokat többféleképpen is ki lehet küszöbölni. Egy egyszerűbb aktív (kamera szempontjából aktív) módszer a kamera által még érzékelt infravörös tartományban fénykibocsátó lámpa használata lehet. A passzív módszer pedig az, amikor a felismerés nem teljes mértékben a fényintenzitásra alapoz, hanem alakfelismerést is alkalmaz. Természetesen a fény egyenletessége az éldetekciót és az alakfelismerést is befolyásolja, továbbá az objektumok színét is.

A másik fontos befolyásolási tényező a nem megfelelő betekintési szög. A kamera megfelelő pozicionálása fontos, mivel a valóságos 3D-s objektumok a kameránál kétdimenziós képpé alakulnak. Rossz szög esetén torzulhat a kapott képünk, bár ez a hátráltató tényező rendelkezik bizonyos tűréssel, és korrigálható, ahogy az 1.3. alfejezetben bemutattuk a perspektívakorrekciónál.

A képfeldolgozás mellett fontos még a közvetítőrendszer is, amely akár internetes közvetítés is lehet. Ez utóbbinál az általános videóközvetítést lehet alapul venni, ahol szerver-kliens struktúrában oldják meg a rendszerek működését. Ha a szerver-kliens közti kommunikációban sávszélességet szeretnénk megtakarítani, akkor a feldolgozott képből csak a lényegi információkat érdemes továbbküldeni. Ezért is fontos, hogy már a küldő oldalon a program fel tudja dolgozni a kamera által közvetített képet. A képfeldolgozásnál a kamera specifikálása, kiválasztása is szerepet játszik. Egyrészről érdemes minél nagyobb felbontásokat használni, hogy jó képminőséget érjünk el. Például egy egyszerű webkamera esetén a 640×480-as VGA-felbontást a kamerák szinte mindegyike megvalósítja (sőt jobb felbontásokat is). Másrészről a képfeldolgozás szempontjából nem túl előnyös túl nagy képekkel, nagy adatmennyiséggel dolgozni.

A videókban való keresés során kifejleszthetők még további, konkrét objektumokat felismerő eljárások. A hangmodalitás is hatékonyabban használható lehet a jövőben, például implementálható a rendszerekbe a hang- és beszédfelismerés, illetve a zene, zaj, természeti hangok, így még jobb képet adva a videótartalmakról, annak gyors áttekinthetőségét és kereshetőségét tovább javítva.

Tanulmányunkban bemutattuk az okos város kameraképeinek elemzéséhez szükséges ismereteket többféle aspektusból. A kamerák által készített felvételek száma – elsősorban az okos város átlagos felhasználói által készített tartalmak (úgynevezett *user generated content*) miatt – napjainkban nagymértékben növekszik. Ezen videók rendszerezése, keresése vagy elemzése jelenleg nem túl hatékony, nem minden esetben megoldott. Az ezzel kapcsolatos feladatokat, problémákat részletesen áttekintettük. A multimédia-tartalmak elemzésénél, más néven a multimédia-bányászatnál a gépi tanulás lehet segítségünkre, így ennek a témakörnek külön fejezetet szenteltünk. A gépi tanulás mellett a képfeldolgozás az, amely nélkül nem lehet a kamerák képeit elemezni. A képek rendelkeznek olyan elemi tulajdonságokkal, amelyek jól jellemezhetik a képet, segíthetik a további feldolgozási lépéseket. Ezt az elemi szintet ragadják meg a képfeldolgozás algoritmusai, amelyeket részletesen

tárgyaltunk. Konklúzióként fontos megemlíteni, hogy az alacsony szinttől a magas szintű megoldásig terjedő hierarchikus építkezésben minden szintnek megvan a maga szerepe, akár a kameraképek alacsony szintű feldolgozását, a számítógépes arcfelismerést vagy a videótartalmak rendszerezését tekintjük. A szöveges tartalmak elemzésére már régebb óta léteznek szolgáltatások, a videótartalmak elemzése, kivonatolása, összegzése azonban nehezebb feladat, és megoldásukra, ahogy azt bemutattuk, más módszerek szükségesek. Az elemzés mellett a keresésre kidolgozott technikákra is külön fejezetet szántunk, mivel a tartalomalapú képvisszakeresés aktívan kutatott téma.

A tanulmányban bemutatott multimédia-területen sok kutatás folyik, és ezek eredményeként mostanában kezd rohamosan nőni a gyakorlati alkalmazások száma, amelyek között még kevés olyat látni, amelyek átfogó módon tudnak sok funkciót megvalósítani. Az okos város környezet azonban egy olyan lehetőség, ahol az átfogó, nagyobb léptékű alkalmazások is megvalósulhatnak, és a multimédia-szolgáltatások más szolgáltatásokkal együtt egy nagyobb rendszerben összekapcsolódva koherens egészet alkothatnak.

8.2. Az okos várossal kapcsolatos médiatechnológiák

Tanulmányunkban részletesen bemutattuk az okos várost érintő multimédia-technológiákat, különös tekintettel a kamerák által készített felvételekre. Ebben az alfejezetben egy nemzetközi kitekintést adunk az okos város témakörében megjelenő technológiai trendekről, az informatikai témájú nemzetközi konferenciákról, rendezvényekről, azokon belül a médiainformatikai témakörre, illetve az okos város – okos közigazgatás alkalmazásokra fókuszálva.

Idén rendezték Velencében a *The Sixth International Conference on Smart Cities, Systems, Devices and Technologies* (SMART 2017, June 25–29.), valamint Indiában az IEEE TENSYMP 2017 – *Technologies for Smart Cities* konferenciákat, ahol láthatóvá vált, hogy a technológiai fejlődés az okos város szolgálatába állítható. 2017. december 11. és 13. között rendezték meg Tajvanban a nemzetközi multimédia-szimpóziumot, az ISM 2017-et, amelyhez egy okos város témájú konferenciát is csatoltak: *1st IEEE International Workshop on Emerging Multimedia Applications and Services for Smart Cities* (with IEEE ISM2017), röviden EMASC 2017. Ennek keretein belül a jövő városának vízióját keresik, amely intelligens városi rendszerek összekapcsolásával képes javítani az életminőséget különböző aspektusokban, mint például közbiztonság, közlekedés, egészségügy, energia.

Ezekhez nagy mennyiségű multimodális szenzor, IoT *(Internet of Things)* és más vezeték nélküli eszközök szükségesek az okos város infrastruktúrájában. A multimédia fontos szerepet tölt be a megvalósításban, ugyanis a városlakók igényei jelentősen megnőttek a multimédia-alkalmazások és szolgáltatások iránt. Ebben az összekapcsolt környezetben a multimodális szenzorok és a közösségi médiát használó emberek által generált multimédia-adatok folytonos *streamje* adja az elemzési lehetőséget, hogy felismerjük a városlakók igényeit, és olyan plusz szolgáltatásokat ajánljunk számukra, amelyekre tényleg szükségük van, és javíthatják életminőségüket.

A multimédia-tartalmak, szenzorok, intelligens eszközök további terjedése segíti az okosváros koncepció-megvalósítását, ugyanakkor kihívások elé állítja az okos város infrastruktúráját és multimédia-technológiáját. A kihívásokra adott válaszokat a legújabb *(state-of-the-art)* technikák, metodikák, multimédia-alkalmazások és szolgáltatások adhatják meg, amelyek segítenek az okos város koncepciójának kibontakozásában, az okos közigazgatás megvalósításában. Ilyen technológiák, eljárások:

- közösségi érzékelés, folyamatkezelés és multimodális adatok aggregálása,
- multimédia-jellegű szituációk kezelése és döntéstámogató rendszerek az okos városhoz,
- multimédia-tartalmak mobilos megosztása az okos városban,
- intelligens multimédia-rendszerek az okos városban,
- *Multimedia Big Data* analitika okosváros-alkalmazások számára,
- IoMT *(Internet of Multimedia Things* vagy más néven *Multimedia IoT)* okos város számára,
- multimédia-típusú sürgősségi válaszadás és eseménykezelés az okos városban,
- multimédia-adatok M2M *(Machine to Machine)* megosztása az okos városban,
- multimédia-szolgáltatás kifejlesztése és menedzselése az okos városban,
- multimédia-biztonság és személyiségvédelem az okosváros-alkalmazásokhoz,
- multimédia-alkalmazások és -szolgáltatások (oktatás, közigazgatás, egészségügy, közlekedés) az okos városban.

8.3. Az okos várost érintő közösségi média kutatása

Az okos városból származó adatok dinamizmusa, heterogenitása és az annotált adatok hiánya kihívások elé állítja a jelenlegi rendszereket. A médiaanalitika kihívásaira adott ötletek és megoldások összegyűjtésére 2017 őszén Szingapúrban rendezték meg a *1st International Workshop on Social Media Analytics for Smart Cities* (SMASC 2017) című workshopot és konferenciát, amely kimondottan az okos város közösségimédia-elemzésével foglalkozott azzal a céllal, hogy feltárja a közösségi média adataiban rejlő lehetőségeket a városi informatika *(urban informatics)* számára. Ehhez a workshophoz más konferenciák és rendezvények is társultak, így az okos várost érintő tömegközlekedés témában megrendezett *Workshop on Big Data Analytics for Enhancing Public Transport (BigTransport17),* valamint a *CIKM 2017 International Conference on Information and Knowledge Management – Smart Cities, Smart Nations.* A szervezők ösztönözték a kutatók közötti kollaborációt az információ-visszakeresés, a közösségimédia-analitika, a számítógépes nyelvészet szakemberei, valamint szociológusok és városi hatóságok között azzal a célzattal, hogy megvalósítható és gyakorlati megoldások szülessenek a városok, városlakók problémáira. Nem titkolt célja volt a workshopnak, hogy bátorítsa a kutatókat olyan technikák kifejlesztésére, amelyek segítségével közösségimédia-adatok elemezhetők és a gyakorlatban alkalmazhatók az okos város informatikai területein, amelyek a következők:

- a városi informatika közösségimédia-forrásainak időbeli, térbeli és szemantikus analízise,
- a város szociálismédia-adatainak vizualizációja idő- és térdimenziók mentén,
- természetes nyelvfeldolgozás és szövegbányászat a városi informatika számára,
- szemantikus webtechnikák a többforrású városi adatokhoz,
- tudás és reláció kinyerése a közösségimédia-forrásokból a városi informatika elemzéséhez,
- zajos közösségimédia-adatok kezelése,
- a közösségi médiához kapcsolódó applikációk a városi informatika számára,
- a közösségi média adatainak gráfanalízise a városi informatikához,
- esettanulmányok a közösségi média adatainak kihasználására a városi életben és a városi tervezésben,
- applikációk a közösségi média polgárai által generált adatokból az okos város számára.

Vákát

Irodalom

Albanese, M. – Fayzullin, M. – Picariello, A. – Subrahmanian, V. S. (2006): The priority curve algorithm for video summarization. *Information Systems*, Vol. 31, No. 7. 679–695.

AlphaGo (2016): *Google's artificial intelligence to take on world champion of ancient Chinese board game.* Australian Broadcasting Corporation. 8 March 2016.

Arman, F. – Depommier, R. – Hsu, A. – Chiu, M. Y. (1994): Content-based browsing of video sequences. *Proceedings of the second ACM international conference on Multimedia*, 97–103.

Bastan, M. – Cam, H. – Gudukbay, U. – Ulusoy, O. (2009): An MPEG-7 compatible video retrieval system with integrated support for complex multimodal queries. IEEE MultiMedia.

Bay, H. – Ess, A. – Tuytelaars, T. – Van Gool, L. (2008): Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, Vol. 110, No. 3. 346–359.

Bay, H. – Tuytelaars, T. – Van Gool, L. (2006): SURF: Speeded up robust features. *Computer vision ECCV 2006*, 404–417.

Beran, V., – Hradiš, M. – Herout, A. – Sumec, S. – Potúček, I. – Zemčík, P. – Chmelaď, P. (2007): Video summarization at Brno University of Technology. *Proceedings of the international workshop on TRECVID video summarization*, 16–19.

Bledsoe, W. W. (1966): *Man-Machine Facial Recognition:* Report on a Large-Scale Experiment. Technical Report PRI 22, Palo Alto, Panoramic Research Inc.

Boreczky, J. S. – Wilcox, L. D. (1998): A hidden Markov model framework for video segmentation using audio and image features. *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on,* Vol. 6. 3741–3744.

Bourel, F. – Chibelushi, C. C. – Low, A. A. (2000): Robust Facial Feature Tracking. *British Machine Vision Conference.* Bristol. 232–241.

Bourlard, H. A. – Morgan, N. (1994): Hidden Markov Models. In Bourlard, H. A. – Morgan, N. eds.: *Connectionist Speech Recognition.* Springer US. 27–58.

Breebaart, J. – McKinney, M. F. (2004): Features for audio classification. In Verhaegh, W F. J. – Aarts, E. – Korst, J. eds.: *Algorithms in Ambient Intelligence.* Springer Netherlands. 113–129.

Canny, J. (1986): A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. 8, No. 6. 679–698.

Chheng, T. (2007): *Video summarization using clustering*. Irvine, University of California, Department of Computer Science.

Dailianas, A. – Allen, R. B. – England, P. (1995): Comparison of automatic video segmentation algorithms. *Proc. SPIE,* Vol. 2615, 2–16.

Farnebäck, G. (2003): Two-frame motion estimation based on polynomial expansion. *Image Analysis*, Vol. 2749. of Lecture Notes in Computer Science. 363–370.

Ffmpeg (s. a.): *Dokumentáció.* Elérhető: www.ffmpeg.org (Letöltés dátuma: 2018. 11. 15.)

Flickner, M. – Sawhney, H. – Niblack, W. – Ashley, J. – Huang, Q. – Dom, B. – Steele, D. (1995): Query by image and video content: The QBIC system. *Computer,* Vol. 28, No. 9. 23–32.

Forney, G. D. (1973): The Viterbi algorithm. *Proceedings of the IEEE,* Vol. 61, No. 3. 268–278.

Hernando, J. (1997): Maximum likelihood weighting of dynamic speech features for CDHMM speech recognition. *IEEE International Conference on Acoustics, Speech, and Signal Processing*. *ICASSP–97. 1997*, Vol. 2, 1267–1270.

Jafri, R. – Arabnia, H. R. (2009): A Survey of Face Recognition Techniques. *Journal of Information Processing Systems,* Vol. 5, No. 2. 41–68.

Jagannathan, L. – Jawahar, C. V. (2005): Perspective correction methods for camera based document analysis. *Proceedings of the First International Workshop on Camera-based Document Analysis and Recognition,* 148–154.

Kalman, R. E. (1960): A new approach to linear filtering and prediction problems. *Journal of Basic Engineering,* Vol. 82, No. 1. 35–45.

Lee, S. – Hayes, M. H. (2002): Real-time camera motion classification for content-based indexing and retrieval using templates. *IEEE International Conference on Acoustics, Speech, and Signal Processing. ICASSP–2002,* Vol. 4, 3664–3667.

Lowe, D. G. (2004): Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision,* Vol. 60, No. 2. 91–110.

Lu, S. – King, I. – Lyu, M. R. (2004): Video summarization by video structure analysis and graph optimization. *IEEE International Conference on Multimedia and Expo, 2004. ICME–'04.* (IEEE ISO 690) Vol. 3, 1959–1962.

Mai, K. – Miller, J. – Zabih, R. (1995): A robust method for detecting cuts and dissolves in video sequences*. Proceeding of ACM Multimedia.* San Francisco.

Maragos, P. – Potamianos, A. – Gros, P. eds. (2008): *Multimodal processing and interaction: audio, video, text.* Vol. 33, Springer Science & Business Media.

Marosvári B. (2016): Annotation of movies for camera movement recognition. *Mesterpróba 2016, Tudományos konferencia távközlés és infokommunikáció témakörében.* Konferenciakiadvány. 69–72.

MediaCollege (s. a.): *Video Camera Tutorials – Shot Types.* Elérhető: www.mediacollege.com/video/shots/ (Letöltés dátuma: 2018. 11. 15.)

Michalski, R. S. – Carbonell, J. G. – Mitchell, T. M. eds. (2013): *Machine learning: An artificial intelligence approach.* Springer Science & Business Media.

Mizsei A. (2016): A jövő elkezdődött – okosházak Budapesten, 2016. december 06., *Archichat blog.* Elérhető: http://archichat.reblog.hu/a-jovo-elkezdodott--okoshazak-budapesten (Letöltés dátuma: 2018. 11. 15.)

Moghaddam, B. – Yang, M. H. (2002): Learning Gender with Support Faces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, 707–711.

Mohan, S. – Avinash, N. – Murali, S. (2008): Rectification of perspective distortion using camera parameters – a perspective geometry based approach. *ICGST-GVIP J*, Vol. 8, No. 1. 1–7.

Moon, T. K. (1996): The expectation-maximization algorithm. *IEEE Signal Processing Magazine*, Vol. 13, No. 6. 47–60.

Mori, S. – Nishida, H. – Yamada, H. (1999): *Optical character recognition.* New York, John Wiley & Sons, Inc.

Morik, K. – Brockhausen, P. – Joachims, T. (1999): Combining statistical learning with a knowledge-based approach: a case study in intensive care monitoring. *16th International Conference on Machine Learning (ICML–99).* San Francisco, CA, USA: Morgan Kaufmann. 268–277.

Najman, L. – Talbot, H. eds. (2013): *Mathematical morphology.* New York, John Wiley & Sons.

Nan, N. – Liu, G. – Qian, X. – Wang, C. (2008): An SVM-based soccer video shot classification scheme using projection histograms. *Advances in Multimedia Information Processing–PCM 2008*, Berlin–Heidelberg, Springer. 883–886.

Natarajan, S. (2009): An efficient video segmentation algorithm with real time adaptive threshold technique. *International Journal of Signal Processing*, Vol. 2, No. 4. 13–28.

Nilesh V. P. – Ishwar K. S. (1997): Video shot detection and characterization for video databases. *Pattern Recognition*, Vol. 30, No. 4. 583–592.

Parr, L. A. (2011): The evolution of face processing in primates. *Philosophical Transactions of the Royal Society B: Biological Sciences*, Vol. 366, No. 1571. 1764–1777.

Pye, D. – Hollinghurst, N. J. – Mills, T. J. – Wood, K. R. (1998): Audio-visual segmentation for content-based retrieval. *5th International Conference on Spoken Language Processing (ICSLP).*

Shinohara, Y. – Otsu, N. (2004): Facial Expression Recognition Using Fisher Weight Maps. *Sixth IEEE International Conference on Automatic Face and Gesture Recognition*, Vol. 100, 499–504.

Sikora, T. (2001): The MPEG-7 visual standard for content description – an overview. *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 11, No. 6. 696–702.

Singh, S. – Papanikolopoulos, N. (1997): *Vision-based detection of driver fatigue*. *Technical report.* University of Minnesota, Department of Computer Science.

Sirovich, L. – Kirby, M. (1987): Low-dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America A.* Vol. 4, No. 3. 519–524.

Sivic, J. – Zisserman, A. (2006): Video Google: Efficient visual search of videos. In Ponce, J. – Herbert, M. – Schmid, C. – Zisserman, A. eds.: *Toward Category-Level Object Recognition.* Berlin–Heidelberg, Springer. 127–144.

Smith, M. A. – Kanade, T. (1998): Video skimming and characterization through the combination of image and language understanding. *Proceedings 1998 IEEE International Workshop on Content-Based Access of Image and Video Database.* 61–70.

Squire, D. M. – Müller, W. – Müller, H. – Pun, T. (2000): Content-based query of image databases: inspirations from text retrieval. *Pattern Recognition Letters,* Vol. 21, No. 13. 1193–1198.

Sural, S. – Qian, G. – Pramanik, S. (2002): Segmentation and histogram generation using the HSV color space for image retrieval. *Processing Proceedings International In Image Conference 2002.* Vol. 2.

Szegedy, C. – Liu, W. – Jia, Y. – Sermanet, P. – Reed, S. – Anguelov, D. Rabinovich, A. et al. (2015): Going deeper with convolutions. *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1–9.

Szegedy, C. – Vanhoucke, V. – Ioffe, S. – Shlens, J. – Wojna, Z. (2016): Rethinking the inception architecture for computer vision. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2818–2826.

Szűcs, G. – Leposa, T. – Turbucz S. (2011): Representative Picture Selection from Albums. *The Third International Conferences on Advances in Multimedia (MMEDIA 2011)*, Budapest, 17–22, April 2011.

Taskiran, C. M. – Amir, A. – Ponceleon, D. B. – Delp, E. J. (2002): Automated video summarization using speech transcripts. *Storage and Retrieval for Media Databases*, Vol. 4676, 371–382.

Turk, M. – Pentland, A. (1991): Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, Vol. 3, No. 1. 71–86.

Vijayarani, S. – Sakila, A. (2015): Multimedia Mining Research – An Overview. *International Journal of Computer Graphics & Animation*, Vol. 5, No. 1. 69.

Viola, P. – Jones, M. (2001): Rapid object detection using a boosted cascade of simple features. *Computer Vision and Pattern Recognition, CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, Vol. 1.

Wu, Y. – Ganapathiraju, A. – Picone, J. (1999): *Baum-welch Re-estimation of Hidden Markov Model.* Mississippi State University, Department of Electrical and Computer Engineering.

Yang, M. H. – Kriegman, D. J. – Ahuja, N. (2002): Detecting faces in images: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, No. 1. 34–58.

Yianilos, P. N. (1993): Data structures and algorithms for nearest neighbour search in general metric spaces. *SODA '93 Proceedings of the fourth annual ACM-SIAM symposium on Discrete algorithms*, Vol. 93, No. 194. 311–321.

Zhang, T. – Kuo, C. C. J. (1998): Content-based classification and retrieval of audio. *Advanced Signal Processing Algorithms, Architectures, and Implementations VIII.* International Society for Optics and Photonics.
Vol. 3461, 432–444.

Zheng, N. – Qi, L. – Guan, L. (2014): Multiple-manifolds discriminant analysis for facial expression recognition from local patches set. *IAPR Workshop on Multimodal Pattern Recognition of Social Signals in Human-Computer Interaction*, Springer. 26–33.

A Dialóg Campus Kiadó a Nemzeti Közszolgálati Egyetem könyvkiadója.

Nordex Nonprofit Kft. – Dialóg Campus Kiadó
www.dialogcampus.hu
www.uni-nke.hu
1083 Budapest, Ludovika tér 2.
Telefon: (30) 426 6116
E-mail: kiado@uni-nke.hu

A kiadásért felel: Petró Ildikó ügyvezető
Felelős szerkesztő: Dalloul Zaynab
Olvasószerkesztő: Bíró Csilla
Tördelőszerkesztő: Gyapjas Anikó
Nyomdai kivitelezés: Pátria Nyomda Zrt.
Felelős vezető: Simon László vezérigazgató

ISBN 978-615-5945-00-7 (nyomtatott)
ISBN 978-615-5945-01-4 (elektronikus)
ISSN 2631-1259

A monográfia az okos város kameraképeinek elemzése során alkalmazott fogalmakat és módszereket tekinti át és rendszerezi. A kötet a médiainformatika korszerű megoldásaira helyezi a hangsúlyt. Különös figyelmet fordít a kameramozgás felismerésére és a kamera által vett mozgó objektumok nyomkövetésére. Önálló fejezetet szentel a mesterséges intelligenciának, azon belül a gépi tanulásnak, amely manapság már az elemzés meghatározó technológiájává vált. Részletesen tárgyalja az okos város polgárai által készített házi videók rendszerezését, a gyors áttekintését támogató számítógépes megoldásokat. A kötetből az is kiderül, hogy a képi tartalom vizsgálata során hogyan lehet megvalósítani az arcdetektálást, a videószegmentálást, hogyan kell készíteni összegző jelenetsort vagy visszakeresni képet tartalom alapján.

A kiadvány a KÖFOP-2.1.2-VEKOP-15-2016-00001
„A jó kormányzást megalapozó közszolgálat-fejlesztés”
című projekt keretében jelent meg.

SZÉCHENYI 2020

Magyarország Kormánya

Európai Unió
Európai Szociális Alap

BEFEKTETÉS A JÖVŐBE