ELEKTROTEHNIŠKI VESTNIK 80(5): 258–265, 2013
IZVIRNI ZNANSTVENI ČLANEK

Hierarhična kompozicionalna arhitektura za detekcijo in razpoznavanje aktivnosti

Janez Perš[1], **Matej Kristan**[1,2], **Rok Mandeljc**[1], **Stanislav Kovačič**[1], **Aleš Leonardis**[2,3]

[1] *Univerza v Ljubljani, Fakulteta za elektrotehniko, Tržaška 25, 1000 Ljubljana, Slovenija*
[2] *Univerza v Ljubljani, Fakulteta za računalništvo in informatiko, Tržaška 25, 1000 Ljubljana, Slovenija*
[3] *University of Birmingham, School of Computer Science and Centre for Computational Neuroscience and Cognitive Robotics, Birmingham B15 2TT, United Kingdom*
E-pošta: janez.pers@fe.uni-lj.si

Povzetek. Večina algoritmov za detekcijo in razpoznavanje aktivnosti temelji na učenju kakovostnih časovno-prostorskih opisnikov. Ti opisniki hitro postanejo prekompleksni, ob tem pa poleg gibanja zajamejo tudi obliko in teksturo, in to na nekontroliran in nepredvidljiv način. Čeprav takšni pristopi v praksi omogočajo visoko uspešnost razpoznavanja, je inferenca počasna, razlage, kako algoritem pride do določenega rezultata, pa večinoma ni mogoče dobiti. V tem članku predstavljamo alternativni pristop, ki temelji na preprostih nizkonivojskih značilnicah, ki zajemajo le gibanje, zaznano v posnetku. Te značilnice s pomočjo hierarhične kompozicionalne sheme v postopku učenja na enem samem kratkem posnetku sestavljamo v vzorce gibanja, t. i. kompozicije. V postopku inference te vzorce poiščemo v analiziranih posnetkih, in jih vzorčimo v "vrečo kompozicij". Za razvrščanje uporabljamo razvrščevalnik SVM z jedrom χ^2. Postopek je računsko učinkovit in primeren za izvedbo na masovno-vzporednih arhitekturah. Zaradi kompozicionalne narave je vzorce gibanja mogoče učinkovito zapisati, učimo pa jih lahko postopoma, sloj za slojem. Postopek omogoča hitro inferenco, končni vektorji značilnic pa so razmeroma nizkodimenzionalni, s čimer dosežemo tudi hitro učenje razvrščevalnika. Predstavljena metoda dosega na standardni bazi UCF Sports najboljše rezultate med metodami, ki temeljijo zgolj na gibanju.

Ključne besede: računalniški vid, razpoznavanje aktivnosti, kompozicionalni modeli

Hierarchical Compositional Architecture for Activity Detection and Recognition

Many activity detection and recognition approaches focus on designing or learning high-quality, low-level spatiotemporal descriptors. These low-level descriptors may become overly complex, encoding motion, shape and texture in uncontrolled and unpredictable ways. While such complexity helps increasing recognition rates, it slows down the process of inference and obscures the reasoning behind the learned descriptors. We present an alternative approach, which is based on primitive features that encode pure motion. These are coupled with a hierarchical scheme to learn motion patterns (compositions) from a single short video. During the inference process, these learned patterns are extracted from the analyzed videos and used with χ^2 SVM classifier in a "bag of compositions" approach. The process is computationally efficient and the method is well-suited for implementation on massively parallel architectures. Due to their compositional nature, motion patterns can be trained incrementally (layer by layer) and stored efficiently. Inference is fast and the final feature vectors are of relatively low dimension, thus enabling fast SVM training. On the standard UCF Sports Action Dataset, the presented method outperforms state-of-the art approaches based on pure-motion.

Prejet 11. oktober, 2013
Odobren 5. december, 2013

1 UVOD

V tem članku je predstavljena hierarhična kompozicionalna shema za detekcijo in razpoznavanje aktivnosti. Kar zadeva razpoznavanje aktivnosti, je gibanje najpomembnejši kanal vizualne informacije, zato predstavljena shema temelji izključno na gibanju. Predstavljeno delo naslavlja tri pomembne izzive pri analizi gibanja. Prvi izziv je količina informacije, saj imamo pri analizi gibanja običajno opravka z zaporedji slik. To pomeni, da je za obdelavo v realnem času treba obdelati do nekaj deset slik na sekundo. Zato izziv ni le, kako zaznati in razpoznati aktivnosti, ampak predvsem, kako to izvesti čim hitreje. Naslednji izziv je, kako razvezati gibanje od drugih vizualnih kanalov (oblike, prostorskega vida in barve) in kljub temu obdržati uporabno informacijo. Takšna zasnova ima dve prednosti: omogoča učinkovitejši zapis, kar zmanjša nevarnost kombinatorične eksplozije zaradi prekompleksnih značilnic, ter olajša razlago, *kako* je sistem prišel do določenega zaključka. To je pomembna prednost v sistemih, ki naj bi izkazovali človeku podobno inteligenco (npr. inteligentni roboti in robotski pomočniki). Po drugi strani pa niso vsi kanali vizualne informacije enako pomembni za vse naloge računalniškega vida (tipičen primer je analiza aktivnosti, kjer je načeloma gibanje najpomembnejši vir

informacij), vendar za dobro razpoznavanje potrebujemo podatke iz več kanalov. Če informacijo združujemo že v zgodnjih stopnjah obdelave, je to lahko manj učinkovito, kot če to počnemo pozneje [14].

Predstavljena metoda naslavlja te izzive s paradigmo kompozicionalne hierarhičnosti, biološko navdihnjenega koncepta načrtovanja algoritmov [9], ki je bil že uspešno uporabljen pri zasnovi najboljših algoritmov za kategorizacijo objektov [15], [3], [20], [8], [2]. V primerjavi s konkurenčnimi pristopi ponujajo kompozicionalne hierarhije učinkovitejšo izrabo obstoječih virov. To dosežejo s pomočjo večkratne uporabe elementov slovarja in tudi že izvedenih izračunov, s tem pa omogočajo tudi učinkovit prenos znanja. Predstavljen pristop sicer v številnih vidikih sledi konceptom, ki so jih predstavili Fidler in drugi [3], [2], vendar pa deluje na gibanju, ne na obliki, in naslavlja problem detekcije in razpoznavanja aktivnosti namesto kategorizacije objektov. Opazovanje le gibanja sicer ne more popolnoma rešiti problema razpoznavanja aktivnosti, vendar lahko, če se osredinimo izključno na gibanje v zgodnji fazi obdelave, združevanje informacij izvedemo pozneje, na učinkovitejši način [14]. S takšnim pristopom tudi ne podvajamo obdelave informacije, ki jo dobro zajamejo že razvite metode, ki naslavljajo druge vizualne kanale, recimo razpoznavanje oblike [3], [2]. Poleg tega, da je to učinkovitejše[14], pa je takšen koncept, torej pozno združevanje, opaziti tudi v bioloških vidnih sistemih [9].

Predlagano shemo prikazujeta sliki 1 in 2. Razumeti je treba, da ob omejitvi zgolj na gibanje v posnetkih ne moremo pričakovati izboljšanja glede na najboljše algoritme, ki vključujejo vse vidne kanale, vključno z obliko in teksturo. Vendar pa je, kot je bilo že omenjeno, ločitev kanalov vidne informacije pomembna pri nadaljnjem razvoju skalabilnih metod razpoznavanja. Čeprav predstavljena metoda uporablja samo gibanje, vseeno dosega presenetljivo dobre rezultate, hkrati pa dosega boljše rezultate kot nekatere metode, ki se zanašajo samo na gibanje brez drugih modalitet.

Preostanek članka je strukturiran, kot sledi. V poglavju 2 so predstavljeni najpomembnejši sorodni pristopi, v poglavju 3 pa podrobnejši opis predlaganega pristopa. V poglavju 4 so navedene izvedbene podrobnosti metode, v poglavju 5 eksperimenti in rezultati, poglavje 6 pa zaključuje članek.

2 SORODNE RAZISKAVE

Raziskovalci so se problema detekcije in razpoznavanja aktivnosti lotevali na različne načine. Večina pristopov temelji na določanju tako imenovanih časovnoprostorskih točk zanimanja (angl. space-time interest points, STIP). Sekvenca slik je v teh primerih obdelana kot tridimenzionalni volumen $I(x, y, t)$, v katerem metode iščejo točke STIP. V okolici teh izračunajo časovnoprostorske opisnike, ki jih potem s procesom rojenja združijo v manjše število *vizualnih besed*. Značilnice so v tem primeru histogrami sopojavljanja teh vizualnih besed, za razvrščanje posnetkov pa je ponavadi uporabljena metoda podpornih vektorjev (angl. support vector machine, SVM).

V [12] so avtorji predlagali tovrstno metodo, ki zajema dve vrsti opisnikov: prva zajame dinamiko scene (torej gibanje), druga pa ozadje (torej statične elemente prizora). V [10] je koncept razširjen z avtomatskim izračunom *optimalnih opisnikov* za dani učni niz posnetkov. Opisnike dobijo s pomočjo globokih nevronskih mrež (angl. deep belief networks). Tako dobijo opisnike, ki opišejo tako časovne kot prostorske spremembe v 3D časovnoprostorskem volumnu.

Še en primer uporabe mešanih modalnosti za analizo aktivnosti [6], kjer je predstavljena hierarhična mreža, enot, občutljivih na gibanje. Metoda izvede zaporedje konvolucij in poišče maksimalen odziv na rezultatih konvolucije. Na vrhu hierarhije je uporabljena metoda SVM za izbiro najbolj diskriminativnih značilnic.

V [7] je predstavljena metoda, ki zakodira lokalno gibanje v shemi, podobni lokalnim binarnim vzorcem. Tako dobi krpice gibanja (angl. motion patches), ki jih uporabi v shemi vreče besed. Tudi [19] uporablja vrečo besed, s tem da identificira majhne atomične dogodke iz premikajočih se slikovnih elementov in jih združi v atomarne aktivnosti. Konceptualno podobna rešitev je [17]. Uporablja gost optični tok ter tako zgradi zapis zaporedij slik s pomočjo gostih trajektorij, ki ga pretvori v opisnike s pomočjo posebnega opisnika (Motion Boundary Histograms, MBH). Metoda dosega odlične rezultate na standardnih zbirkah podatkov, še boljše rezultate pa dosega, če je kombinirana z informacijo o obliki. Izvedba, ki uporablja izključno MBH na gostih trajektorijah, je ena redkih metod, ki temelji izključno na uporabi informacije o gibanju.

Metode poskušajo izboljšati delovanje na različne načine. Tako recimo [5] razširi prostorski model vreče besed z diskriminativno razširitvijo, ki omogoča sočasno segmentacijo in lokalizacijo dogodkov. V [1] je predlagano še bolj intenzivno mešanje modalitet za lokalizacijo aktivnosti – sliko zakodira kot polje časovnoprostorskih opisnikov in poišče časovnoprostorski podgraf, ki maksimira odziv naučenih razvrščevalnikov.

V nasprotju z večino pristopov [16] ugotavlja, da aktivnost ni nujno povezana s kontekstom in je mogoče dobiti boljše rezultate, če jo obravnavamo ločeno. Morda najbolj soroden pristop k predlaganemu v tem članku je [4]. Metoda zazna Harrisove značilne točke ločeno v prostorskih in časovnoprostorskih ravninah na različnih skalah. S pomočjo enostavnega prostorskega kodiranja hierarhično gradi predstavitve, tako da se najpogostejši vzorci prenašajo na naslednji sloj. Metoda doseže odlične rezultate na standardnih bazah posnetkov, vendar pa gre, čeprav avtorji namigujejo drugače, tudi tu za primer mešanja modalitet – predstavitev, ki jo zgradi takšna metoda, zajame tako gibanje kot obliko in ni nikakršnega zadržka, da ob ustreznih učnih podatkih ne

bi zgradila hierarhije, ki bi bila popolnoma oprta samo na obliko in teksturo.

3 STRUKTURA KOMPOZICIONALNEGA HIERARHIČNEGA MODELA

Predlagani model za analizo gibanja je sestavljen iz treh stopenj obdelave podatkov, kot to prikazuje slika 1. Namen najnižje stopnje, sestavljene iz slojev L^0 in L^1, je določanje najpreprostejših značilnic gibanja kvantiziranih vektorjev optičnega toka. Srednja stopnja predstavlja hierarhično kompozicionalno strukturo in je sestavljena iz več slojev, od L^2 do L^N. Večslojna struktura zmanjšuje kompleksnost učenja zaradi omejenega zaznavnega polja, v katerem opazujemo sosede vsakega zaznanega elementa, saj izvajamo učenje le na enem sloju hkrati. Takšna shema odpravlja potrebo po sočasni oceni velikega števila parametrov v fazi učenja – najprej izvedemo učenje na najnižjem sloju, potem pa nadaljujemo s slojem nad njim. Najvišja stopnja modela vsebuje diskriminativno komponento – razvrščevalnik SVM. Vsak sloj L^i ima svoj slovar, Λ^i. Λ^1 je določen vnaprej, medtem ko slovarje višjih slojev pridobimo v postopku učenja.

3.1 Izračun gibanja, sloja L^0 in L^1

Prva stopnja obdelave poskrbi za izračun preprostih elementov gibanja iz posnetkov. L^0 zgradi Gaussovo piramido vsake slike ter izračuna razliko Gaussov, podobno kot [11]. Tako nastanejo slike tistih področij, kjer je gibanje mogoče dobro zaznati (robovi, oglišča), in to na več skalah. Po postopku dušenja nemaksimalnih vrednosti (angl. non-maxima suppression, NMS) je vsako od teh področij predstavljeno z eno samo točko. Velikost okna, v katerem izvajamo dušenje, in minimalna vrednost preživelega maksimuma sta parametra sloja L^0.

Sloj L^1 primerja detektirane točke na dveh zaporednih slikah in ocenjuje lokalno gibanje s pomočjo iskanja najbližjega soseda v lokalni okolici. Koraki na slojih L^0 in L^1 so ponovljeni na vseh skalah piramide, s tem pa lahko metoda zazna gibanje v velikem razponu magnitud. Kot končni korak obdelave na L^1 so vektorji gibanja kvantizirani glede na njihovo smer in magnitudo – števili mogočih amplitud in smeri predstavljata parametra sloja L^1. Kvantizacija je izvedena tako, da so vektorji z dolžino nič izločeni iz nadaljnje obdelave. Število mogočih smeri in magnitud določa velikost slovarja na sloju L^1 – v našem primeru osmih mogočih smeri in treh mogočih magnitud je velikost slovarja $8 \times 3 = 24$. Od te stopnje naprej se vsa obdelava izvaja na kvantiziranih vektorjih gibanja. Rezultati slojev L^0 in L^1 so prikazani na sliki 2, skupaj s 24 osnovnimi elementi fiksnega slovarja sloja L^1.

3.2 Dušenje nezaželene vizualne informacije

Posebnost prve stopnje predlagane strukture je dušenje kakršnekoli druge vizualne informacije razen gibanja. Z dušenjem nemaksimalnih vrednosti na sliki razlike Gaussov učinkovito odstranimo vso informacijo v okolici ohranjene točke, tako da algoritem ne zajame nikakršnih informacij o obliki ali teksturi. Tako v nasprotju z nekaterimi drugimi pristopi, na primer [4], [10], preprečimo, da bi kompozicionalna struktura na drugi stopnji obdelave gradila kakršnekoli modele oblike.

3.2.1 Grajenje hierarhičnih kompozicij – sloji L^2-L^N: Kompozicionalna hierarhična struktura je sestavljena iz N identično strukturiranih slojev, pri čemer je N parameter strukture. Vhodni podatki v strukturo so kvantizirani in labelirani vektorji gibanja. Vsak vektor V je določen s svojimi tremi časovnoprostorskimi koordinatami in svojo oznako l, $V = V(x, y, t, l)$. Izhod vsakega sloja L^i v kompozicionalni strukturi predstavljajo kompozicije parov elementov nižjega sloja $L^{(i-1)}$. Vsaka od kompozicij predstavlja nov element P, $P = P(x, y, t, l)$, in predstavlja načeloma enega od vhodnih elementov višjega sloja. Vhodne elemente v najnižji kompozicionalni sloj L^2 predstavljajo kar kvantizirani vektorji, zato zanje velja $P^1 = V(x, y, t, l^1)$, kjer oznaka elementa l^1 zavzema eno od mogočih vrednosti iz slovarja Λ^1, določenih s kvantizacijo na L_1, $l^1 \in \Lambda^1$. Slovarje višjih slojev, $\Lambda^i, i > 1$ pridobimo s postopkom učenja.

Kompozicije na vseh slojih $L^i, i > 1$ so definirane kot pari tesno sopojavljenih elementov iz nižjega sloja, torej velja za element j iz sloja L^i naslednje:

$$P^i_j = P^{i-1}_m \cup P^{i-1}_n, \tag{1}$$

pri čemer sta m in n indeksa dveh elementov iz nižjega sloja, ki tvorita element P^i_j. Časovnoprostorske koordinate kompozicije so definirane kot centroidi lokacij obeh elementov, ki sestavljata kompozicijo:

$$x^i_j = \frac{x^{i-1}_m + x^{i-1}_n}{2}, y^i_j = \frac{y^{i-1}_m + y^{i-1}_n}{2}, t^i_j = \frac{t^{i-1}_m + t^{i-1}_n}{2}, \tag{2}$$

medtem ko je oznaka l^i_j določena skozi dvodimenzionalno indeksno tabelo $\psi^{i-1,i}$:

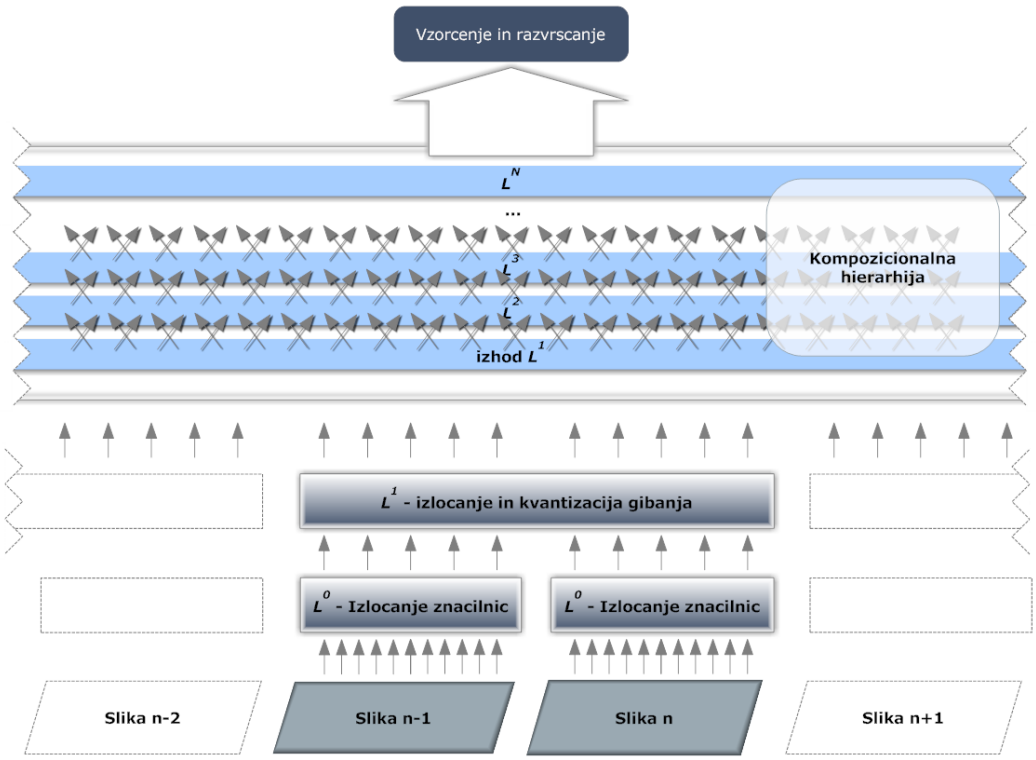
$$l^i_j = \psi^{i-1,i}(l^{i-1}_m, l^{i-1}_n), \tag{3}$$

ki indeksira pare oznak iz sloja L^{i-1} v oznake na sloju L^i. Ničle v indeksni tabeli $\psi^{i-1,i}$ označujejo tiste kompozicije, ki jih v indeksiranju iz sloja v sloj zavržemo.

Tesna sopojavitev dveh elementov je določena s pomočjo dveh pragov, R^i_{rfxy}, na prostorski evklidski razdalji med obema elementoma P^{i-1}_m in P^{i-1}_n, in R^i_{rft} na njuni razdalji vzdolž časovne osi:

$$\sqrt{(x^{i-1}_m - x^{i-1}_n)^2 + (y^{i-1}_m - y^{i-1}_n)^2} < R^i_{rfxy}. \tag{4}$$

$$t^{i-1}_m - t^{i-1}_n < R^i_{rft}. \tag{5}$$



Slika 1: Tri stopnje obdelave v predlaganem kompozicionalnem hierarhičnem modelu. Prva stopnja je sestavljena iz L^0 in L^1, druga iz (N-1) slojev L^2 - L^N, v tretji stopnji pa sta vzorčenje (vreča kompozicij) in razvrščevalnik SVM.

Oba pragova R^i_{rfxy} in R_{rft} odražata velikost cilindrično oblikovanega lokalnega *zaznavnega polja* in tako predstavljata parametra vsakega od kompozicionalnih slojev L^i. V kompozicionalni shemi naj bi velikost zaznavnega polja povečevali iz sloja na sloj, s čimer lahko shema gradi vedno kompleksnejše predstavitve vhodnih podatkov – v našem primeru gibanja. Upoštevati je treba, da predlagana shema ne kodira časovnoprostorskih odnosov med zaznanima elementoma (prej, pozneje, spodaj, zgoraj), zato ne modelira ničesar drugega kot enostavno bližino dveh elementov. Zato ima omejeno moč predstavljanja kompleksnih struktur, kot bi načeloma bila oblika na podlagi gibanja (angl. shape from motion).

3.2.2 Učenje strukture v slojih L^2-L^N: Za učenje sloja L^i potrebujemo izhod nižjega sloja L^{i-1}, ki je lahko kompozicionalen $(L^i, i > 1)$ ali pa ne (L^1). Učenje modela na sloju L^i je izvedeno s pomočjo optimizacije indeksne tabele $\psi^{i-1,i}$ glede na naslednja dva kriterija:

1) Pokritost: Čim več osnovnih elementov iz sloja L^1 (z dna kompozicionalne strukture) naj preživi v prehodu iz sloja L^{i-1} v sloj L^i. Ta kriterij zagotavlja, da potencialno uporabna informacija, ki vstopa v kompozicionalno strukturo, ni po nepotrebnem zavržena.
2) Slovar sloja, ki ga učimo, L^i, Λ_i naj je čim manjši.

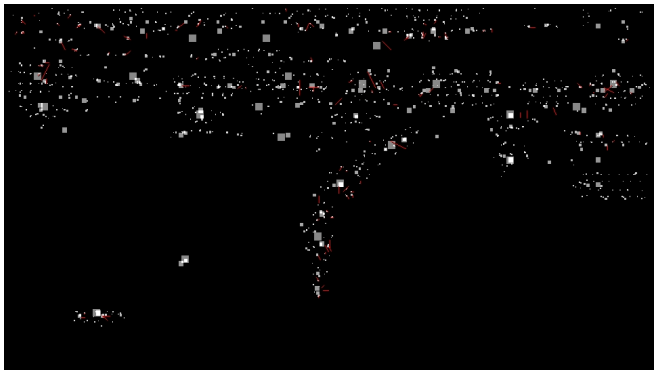
Omenjena kriterija sta v nasprotju in težita k slovarju, ki je kompromis med najboljšo rekonstrucijo vhodne informacije in najkrajšim naborom simbolov. Učenje je izvedeno kot nelinearna optimizacija naslednje kriterijske funkcije:

$$f(x) = -C^1 + \alpha|\Lambda^i|, \tag{6}$$

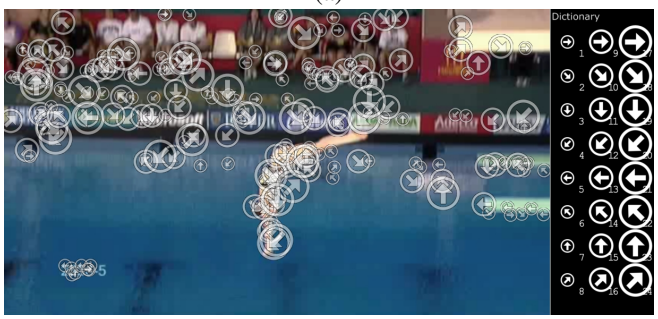
kjer je C^1 pokritost (razmerje števila osnovnih elementov iz sloja L^1, katerih kompozicije na sloju L^i bi preživele ob trenutnem slovarju, in števila vseh elementov, ki iz L^1 vstopajo v kompozicionalno strukturo), $|\Lambda^i|$ is je število elementov slovarja, α pa je utež, s katero nadzorujemo intenzivnost regularizacije, torej točko kompromisa med nasprotujočima si kriterijema. Domena x so vse mogoče kombinacije ničelnih in neničelnih elementov v indeksni tabeli $|\Lambda^i|$. S pomočjo optimizacije v bistvu določimo takšen slovar Λ^i, da velja:

$$\Lambda^i = \arg\min_x f(x). \tag{7}$$

Očitno je, da bi v najslabšem primeru popolnoma naključnih vhodnih podatkov število kompzicionalnih elementov v slovarju $|\Lambda^i|$ raslo s kvadratom i, vendar pa se izkaže, da pri realističnih podatkih raste počasneje. Da pospešimo postopek optimizacije in bistveno zmanjšamo prostor iskanja optimalnega slovarja, lahko že pred postopkom optimizacije odstranimo delež mogočih kompozicij, ki vstopajo v optimizacijski proces. To izvedemo le za učenje sloja L^3 (in višjih), kjer tako odstranimo 1% mogočih kompozicij, ki se v podatkih pojavljajo najredkeje.



(a)



(b)

Slika 2: Značilnice gibanja na primeru posnetka iz kategorije "Diving", zbirke podatkov UCF Sports Action Dataset [13]. (a) Izhod sloja L^0, kvadratki označujejo lokalne maksimume razlike Gaussov. Večji kvadratek pomeni bolj grobo skalo. Črte označujejo vektorje gibanja. (b) Izhod sloja L^1 in fiksen slovar tega sloja. Radiji krogov ustrezajo amplitudi zaznanega gibanja, puščice pa kažejo smer, oboje že po postopku kvantizacije.

3.2.3 Inferenca na slojih L^2-L^N: Inferenca je izvedena kot zaporedje indeksiranj skozi indeksne tabele $\psi^{i-1,i}$. Trenutno metoda ne upošteva drugih časovnoprostorskih odnosov med elementi, razen tega da določen element leži znotraj (časovnoprostorskega) zaznavnega polja nekega drugega elementa. Zato je indeksiranje s stališča računske zahtevnosti izjemno hitra operacija, takoj ko potrdimo sopojavitev dveh elementov. Iskanje sopojavitev je načeloma računsko potratna operacija, vendar pa je zaradi počasne (manj kot kvadratične) rasti velikosti slovarjev $|\Lambda^i|$ skozi sloje inferenca še učinkovitejša, če iščemo sopojavitve samo tistih elementov, ki bi preživeli indeksiranje skozi dano indeksno tabelo Λ^i.

3.3 Zajem značilnic in razvrščanje

Osnovni elementi, ki jih zaznamo pri obdelavi zaporedja slik skozi sloja L^0 in L^1, predstavljajo osnovne gradnike gibanja, ki pa zrastejo v vedno kompleksnejše gradnike skozi višje sloje. Pojavljanje elementov skozi hierarhično strukturo je mogoče vzorčiti na različne načine. Predlagana metoda zgradi *vrečo kompozicij* tako, da izračuna histograme pojavljanja vseh elementov na vseh slojih in jih sestavi v en sam vektor značilnic. Ta postopek je na vrhu predlagane kompozicionalne piramide, kot jo prikazuje slika 1.

Pristop z vrečo kompozicij naredi predlagano metodo povsem invariantno na časovni odmik aktivnosti znotraj zaporedja slik. Da se ta učinek zmanjša, so podatki najprej razdeljeni vzdolž časovne osi v M ločenih, enako dolgih odsekov in zajeti v M histogramov, ki so na koncu sestavljeni v en sam daljši vektor značilk. Za razvrščanje tako pridobljenih vektorjev značilk je bila uporabljena javno dostopna izvedba razvrščevalnika [10], katere struktura je opisana v [18], temelji pa na metodi podpornih vektorjev z jedrom χ^2.

3.4 Velikost naučenega modela

V predstavljeni arhitekturi je znanje zgoščeno v hierarhiji indeksnih tabel $|\lambda^i|$, ki omogočajo prevedbo oznak elementov na sloju l^{i-1} v oznake na sloju l^i. Tako za slovarje z do 256 elementi potrebujemo le polje osembitnih števil z maksimalno dimenzijo 256×256, da shranimo eno indeksno tabelo, pa še v tej tabeli ima velik del elementov vrednost nič. Če je kot zadnja stopnja razpoznavanja uporabljen razvrščevalnik SVM, je treba shraniti še njegov model, podobno kot pri drugih pristopih z vrečo besed, kot sta [10] in [18].

4 IZVEDBA PREDLAGANE METODE

Predlagana shema izraža najpomembnejši prednosti kompozicionalnih hierahičnih arhitektur – hitro inferenco in kompakten zapis znanja. Toda, čeprav so kompozicionalne arhitekture znane kot hitre in učinkovite pri obdelavi posameznih slik, se pri obdelavi posnetkov

soočamo s problemom, katerega računska zahtevnost je za nekaj razredov velikosti večja. Shema je bila načrtovana z namenom vzporedne izvedbe, predvsem na modernih splošnonamenskih grafičnih procesorjih (angl. (General Purpose) Graphic Processing Unit, GPU). Vsi najpomembnejši sestavni deli predlagane metode se tako zlahka preslikajo na masivno vzporedno računsko arhitekturo, kot je to prikazano v nadaljevanju.

4.1 Sloja L^0 in L^1

Čeprav je mogoče Gaussovo glajenje zlahka izvesti na grafičnem procesorju, le-to ni glavno ozko grlo. Pomembnejše je, da sta tako dušenje nemaksimalnih vrednosti in izračun vektorjev gibanja (iskanje korespondenc) izvedena na masivno vzporedni arhitekturi, s pomočjo orodja Jacket za Matlab* in njegovega programskega bloka `gfor`. Za boljši izkoristek grafičnega procesorja slike obdelujemo v paketih po 20 slik.

4.2 Sloji L^2 in višji

Čeprav hierarhična shema omogoča učinkovito iskanje s tem, da iščemo samo tiste kompozicije, ki bodo preživele indeksiranje, je mogoče vse razdalje med elementi hitro izračunati že vnaprej z uporabo grafičnega procesorja. Tako se proces bistveno pospeši. Po drugi strani pa so tudi vektorji značilnic razmeroma nizkodimenzionalni, kar omogoča hitro učenje razvrščevalnika SVM in hitro razpoznavanje.

4.3 Učenje

Računsko najbolj kompleksen del algoritma je optimizacija kriterijske funkcije (6) v postopku učenja posameznega sloja. Upoštevati je treba, da zahteva izračun pokritosti C v vsaki iteraciji na relativno velikih količinah vhodnih podatkov. Za optimizacijo je trenutno uporabljen genetski algoritem, ki se dobro skalira na večjedrnih procesorjih osebnih računalnikov.

5 EKSPERIMENTI IN REZULTATI

Predstavljeno shemo smo ocenjevali po treh kriterijih: uspešnost razpoznavanja in detekcije, učinkovitost računanja in velikost modela. Vsi eksperimenti so bili izvedeni na osebnem računalniku s štirijedrnim procesorjem Intel Xeon E5-1620 s taktom 3.60 GHz. Kot grafični procesor je bila uporabljena grafična kartica Nvidia Quadro 2000 z 1 GB videopomnilnika. Uporabili smo Matlab 2011a na Linuxu (distribucija Fedora 18) in Jacket 2.2 za obdelavo na grafičnem procesorju. Za nelinearno optimizacijo smo uporabili genetski algoritem iz Matlabove orodjarne. V eksperimentih smo parametre algoritma nastavili na naslednje vrednosti: velikost okna za dušenje nemaksimalnih vrednosti je bila 7 slikovnih elementov, velikost zaznavnega polja je bila nastavljena na $R^2_{rfxy} = 3.5$ in $R^3_{rfxy} = 7$ slikovnih elementov, R^i_{rft} pa je bil nastavljen na 0 za vse sloje; s tem je bila hierarhija prisiljena v učenje kompozicij samo v prostorski (XY) ravnini. Parameter α je bil nastavljen na 0.6. Parametri so bili določeni eksperimentalno — pazili smo, da sta sloja L^0 in L^1 v sliki detektirala dovolj osnovnih elementov gibanja, da je bila gradnja kompozicij sploh mogoča, in da je bilo učenje izvedeno v razumnem času ter z razumno porabo pomnilnika.

Za učenje kompozicionalne strukture smo uporabili samo en video z dolžino 480 slik. Video ni nikakor povezan z nobeno od testnih baz in zajema 24 odsekov otroške risanke s po 20 slikami. Slika 3 prikazuje tri tipične slike iz učnega posnetka.

Testiranje je bilo izvedeno na dveh standardnih in široko uporabljanih zbirkah posnetkov: *UCF Sports Action Dataset* [13] *Holywood2 Dataset* [12], [18]. Prva vsebuje 10 kategorij športnih aktivnosti, medtem ko druga vsebuje 12 kategorij vsakodnevnih aktivnosti, pridobljenih iz zbirke znanih holywoodskih filmov. Ker je učni video imel 384×288 slikovnih elementov, smo vse testne posnetke prevzorčili na vertikalno ločljivost 288 slikovnih elementov in ohranili razmerje med višino ter širino slike. Tako smo ločljivost vseh posnetkov iz prve zbirke in večine posnetkov iz druge zbirke zmanjšali, vendar pa je zaradi velikih razlik v ločljivosti posnetkov iz baze Holywood2 prišlo do tega, da se je ločljivost nekaterih posnetkov s tem povečala.

Uspešnost razpoznavanja smo ovrednotili po metodologijah, ki sta predpisani za vsako od zbirk. V skladu s tem za UCF Sports Action Dataset navajamo povprečno točnost, za Holywood2 pa povprečno natančnost.

5.1 Uspešnost razpoznavanja

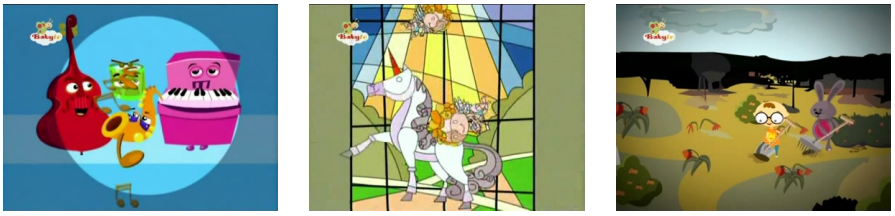
Uspešnost razpoznavanja navajamo glede na število uporabljenih kompozicionalnih slojev. Zaradi nepristranske primerjave navajamo tudi rezultate, ki smo jih dobili z neposrednim zajemom sloja L^1 brez kompozicionalne arhitekture. Rezultati za UCF Sports Action Dataset so prikazani v tabeli 1. Učinkovitost naše sheme se ni povečala z dodajanjem tretjega sloja, zato smo se omejili samo na strukturo s slojema L^1 in L^2.

Naša metoda (zajeti sloji)	Povp. točnost
L^1	73.3%
L^1+L^2	**80.0**%
Najboljši objavljeni rezultati	
(kombinacija značilnic) [17]	88.2%
(samo gibanje) [17]	75.2%

Tabela 1: Rezultati na zbirki UCF Sports Action dataset, primerjani z najboljšimi objavljenimi rezultati. Predlagana metoda je boljša od [17] – izvedbe, ki upošteva samo trajektorije gibanja, brez drugih vizualnih informacij.

Vidimo, da tudi v konfiguraciji $L^1 + L^2$ naša metoda deluje bolje kot najboljša objavljena metoda [17]. Upoštevati je treba, da so metode, ki upoštevajo izključno gibanje, sorazmerno redke, in da veliko avtorjev

*http://wiki.accelereyes.com/wiki/index.php?title=Main_Page



Slika 3: Tri tipične slike z učnega posnetka dolžine 480 slik, s katerim smo učili kompozicionalni del hierarhije

svoje metode uvršča v to kategorijo, čeprav upoštevajo tudi druge kanale vizualne informacije (npr. teksturo in obliko). Glede na to, da UCF Sports Action Dataset vsebuje veliko lepo strukturiranega gibanja, rezultati niso presenetljivi. Rezultati za zbirko Holywood2 so prikazani v tabeli 2.

Naša metoda (zajeti sloji)	Povp. natančnost
L^1	28.9%
L^1+L^2	32.3%
L^1+L^2+L^3	33.1%
Najboljši objavljeni rezultati	
(kombinacija značilk) [17]	58.3%
(samo gibanje) [17]	47.7%
SIFT+HoG+HoF (2009) [12]	32.6%

Tabela 2: Rezultati na zbirki posnetkov Holywood2 dataset in primerjava z najboljšimi objavljenimi rezultati ter rezultati, ki so jih dobili avtorji zbirke Holywood2.

Rezultati na bazi Holywood2 so slabši, čeprav so primerljivi z rezultati, ki so jih dobili avtorji zbirke Holywood2 [12].

5.2 Računska učinkovitost

Tabela 3 prikazuje porabljen čas na sliko, ki ga je porabila predlagana metoda v vsaki od stopenj obdelave podatkov med obdelavo najdaljšega posnetka iz zbirke Holywood2.

Opravilo	Čas/sliko
*Razlika Gaussov	28 ms
*NMS	158 ms
*Ocena vektorjev gibanja	12 ms
Kvantizacija	1 ms
*Inferenca sloj-sloj	22 ms

Tabela 3: Čas, porabljen v vsaki od stopenj obdelave, na sliko. Testirano na najdaljšem posnetku iz baze Holywood2, ki vsebuje 2958 slik. Opravila, označena z zvezdico (*), so tekla na grafičnem procesorju.

Tabela 4 prikazuje čas, potreben za optimizacijo v fazi učenja, ter čas, potreben za evalvacijo z metodo podpornih vektorjev (učenje razvrščevalnika in testiranje). Časi so prikazani za celoten učni posnetek in celotno zbirko podatkov. Navajamo tudi dimenzionalnost značilnic, ki so predstavljale vstopne podatke za metodo podpornih vektorjev (SVM).

Opravilo	Skupni čas	Dimenzionalnost
*Optimizacija, L^2	8 min	/
*Optimizacija, L^3	45 min	/
SVM, L^1	2.9s	120
SVM, $L^1 + L^2$	8.8s	740
SVM, $L^1 + L^2 + L^3$	43.2s	5686
SVM, [10]	31.9s	3000

Tabela 4: Približni časi, ki jih je porabil genetski algoritem za optimizacijo med učenjem (480 slik), časi, potrebni za učenje in testiranje po metodi podpornih vektorjev (SVM) za celotno zbirko Holywood2, ter dimenzionalnost značilnic, na katerih deluje SVM. Opravila, označena z zvezdico (*), so tekla na vseh štirih jedrih procesorja vzporedno.

5.3 Učinkovitost zapisa znanja

Navajamo še količino pomnilnika, potrebno za zapis naučene kompozicionalne hierarhije. Struktura je shranjena v eni ali več indeksnih tabelah, $\psi^{i-1,i}$. Tabele vsebujejo veliko ničel iz dveh razlogov: nekatere sicer mogoče kompozicije se nikoli ne pojavijo v učnih podatkih, ali pa jih zavrže postopek optimizacije. Za preizkus smo v datoteko zapisali vsebino tabel v obliki 8- ali 16-bitnih podatkov, in jih učinkovito kodirali s pomočjo orodja `gzip`. Rezultate prikazuje tabela 5.

Tabela	Dimenzije	Bajtov	Bajtov, gzip
$\psi^{1,2}$	24×24, 8 bit	576	166
$\psi^{2,3}$	74×74, 16 bit	10 952	1112

Tabela 5: Dimenzije indeksnih tabel in njihova velikost pred učinkovitim zapisom s pomočjo orodja gzip in po njem.

6 SKLEP

Rezultati predlagane metode so obetavni, čeprav je jasno, da gibanje ne vsebuje vseh potrebnih informacij za najboljše razpoznavanje aktivnosti. Odlični rezultati na zbirki posnetkov UCF Sports Action Dataset kažejo, da se metoda odreže tam, kjer opazovana aktivnost dominira v prizoru, gibanje v prizoru pa je za aktivnost značilno. Slabši rezultati na zbirki posnetkov Holywood2 imajo več razlogov: aktivnosti v zbirki so močno

odvisne od konteksta, gibanje samo pa v večini primerov iz te zbirke ne ponuja dovolj informacije za dobro razpoznavo aktivnosti. Zato imajo metode, ki zajamejo širok spekter vidne informacije, čeprav nenadzorovano, na tej zbirki prednost. Ugotovimo lahko tudi, da nobena od teh zbirk ne vsebuje zelo kompleksnih vzorcev gibanja, zato je za razpoznavanje potrebnih le malo število slojev kompozicionalne hierarhije. Predstavljena metoda sicer ne izkorišča možnosti masivno vzporednega računanja tako učinkovito kot metode, ki uporabljajo konvolucijo za detekcijo značilnih točk v časovnoprostorskih volumnih posnetkov, vendar je mogoče na vzporednih računskih arhitekturah izvesti tako rekoč vse časovno potratne korake, ob tem pa izkoristiti tudi prednosti, ki jih prinaša hierarhično-kompozicionalna zasnova. V nadaljnjem delu bomo predstavljeno metodo združili s sorodnimi kompozicionalnimi pristopi za detekcijo oblik in tako še izboljšali uspešnost razpoznavanja.

ZAHVALA

To delo je nastalo v okviru raziskovalnega projekta J2-4284, raziskovalnih programov P2-0095, P2-0214 in P2-0098, ter pogodbe o financiranju št. 1000-10-310118, ki jih je vse financirala Javna agencija za raziskovalno dejavnost Republike Slovenije.

LITERATURA

[1] C.-Y. Chen in K. Grauman. Efficient activity detection with max-subgraph search. V *CVPR*, str. 1274–1281. IEEE, 2012.
[2] S. Fidler, M. Boben, in A. Leonardis. Evaluating multi-class learning strategies in a hierarchical framework for object detection. V *Neural Inf. Proc. Systems*, 2009.
[3] S. Fidler in A. Leonardis. Towards scalable representations of object categories: Learning a hierarchy of parts. V *CVPR*, str. 1–8, 2007.
[4] A. Gilbert, J. Illingworth, in R. Bowden. Action recognition using mined hierarchical compound features. *TPAMI*, 33(5):883–897, 2011.
[5] M. Hoai, Z.-Z. Lan, in F. De la Torre. Joint segmentation and classification of human actions in video. V *CVPR*, str. 3265–3272. IEEE, 2011.
[6] H. Jhuang, T. Serre, L. Wolf, in T. Poggio. A biologically inspired system for action recognition. V *ICCV*, str. 1–8, 2007.
[7] O. Kliper-Gross, Y. Gurovich, T. Hassner, in L. Wolf. Motion interchange patterns for action recognition in unconstrained videos. V *ECCV*, str. 256–269. Springer, 2012.
[8] I. Kokkinos in A. Yuille. Inference and learning with hierarchical shape models. *Int. J. Comput. Vision*, 93(3):1–25, 2011.
[9] N. Kruger, P. Janssen, S. Kalkan, M. Lappe, A. Leonardis, J. Piater, A. Rodriguez-Sanchez, in L. Wiskott. Deep hierarchies in the primate visual cortex: What can we learn for computer vision? *TPAMI*, PP(99):1–1, 2012.
[10] Q. Le, W. Zou, S. Yeung, in A. Ng. Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. V *CVPR*, str. 3361–3368, 2011.
[11] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, 2004.
[12] M. Marszalek, I. Laptev, in C. Schmid. Actions in context. V *CVPR*, str. 2929–2936, 2009.
[13] M. Rodriguez, J. Ahmed, in M. Shah. Action mach a spatio-temporal maximum average correlation height filter for action recognition. V *CVPR*, str. 1–8, 2008.
[14] K. F. Shahbaz, R. Anwer, J. van de Weijer, A. Bagdanov, M. Vanrell, in A. Lopez. Color attributes for object detection. V *Comp. vis. patt. recognition*, 2011.
[15] S. Todorovic in N. Ahuja. Learning subcategory relevances for category recognition. V *CVPR*, str. 1–8, 2008.
[16] E. Vig, M. Dorr, in D. Cox. Space-variant descriptor sampling for action recognition based on saliency and eye movements. V *ECCV*, volume 7578 of *Lecture Notes in Computer Science*, str. 84–97. 2012.
[17] H. Wang, A. Klaser, C. Schmid, in C.-L. Liu. Action recognition by dense trajectories. V *CVPR*, str. 3169–3176. IEEE, 2011.
[18] H. Wang, M. M. Ullah, A. Kläser, I. Laptev, in C. Schmid. Evaluation of local spatio-temporal features for action recognition. V *Proc. British Machine Vision Conference*, str. 127, sep 2009.
[19] G. Zen in E. Ricci. Earth mover's prototypes: A convex learning approach for discovering activity patterns in dynamic scenes. V *CVPR*, str. 3225–3232. IEEE, 2011.
[20] L. L. Zhu, Y. Chen, A. Torralba, W. Freeman, in A. Yuille. Part and appearance sharing: Recursive compositional models for multi-view multi-object detection. V *CVPR*, 2010.

Janez Perš je doktoriral leta 2004 na Fakulteti za elektrotehniko Univerze v Ljubljani. Trenutno je docent na Fakulteti za elektrotehniko. Raziskovalno deluje v okviru Laboratorija za strojni vid, ukvarja pa se s področjem računalniškega vida in analize zaporedij slik, sledenjem objektov, analizo človeškega gibanja, dinamično biometrijo ter z avtonomnimi in porazdeljenimi sistemi.

Matej Kristan je leta 2008 prejel naziv doktor znanosti s področja elektrotehnike na Fakulteti za elektrotehniko Univerze v Ljubljani. Trenutno je docent na Fakulteti za računalništvo in informatiko ter na Fakulteti za elektrotehniko na isti univerzi. Njegovo raziskovalno področje obsega verjetnostne metode računalniškega vida in strojnega učenja s poudarkom na vizualnem sledenju, verjetnostnih dinamičnih modelih, sprotnem učenju in mobilni robotiki.

Rok Mandeljc je diplomiral leta 2010 na Fakulteti za Elektrotehniko v Ljubljani, kjer je trenutno zaposlen kot mladi raziskovalec v Laboratoriju za strojni vid. Njegovo glavno raziskovalno področje zajema metode računalniškega vida, obdelave slik ter zlivanja informacije z različnih senzorjev za potrebe detekcije, lokalizacije in sledenja oseb v prostoru.

Stanislav Kovačič je redni profesor na Fakulteti za elektrotehniko Univerze v Ljubljani in predstojnik Laboratorija za strojni vid. Kot gostujoči raziskovalec je deloval v laboratoriju GRASP na University of Pennsylvania, Technische Fakultät der Friedrich–Alexander–Universität v Erlangu in na Fakulteti za elektrotehniko in računalništvo Univerze v Zagrebu. Njegovo raziskovalno delo zajema različne vidike analize slik in videa z aplikacijami v medicini, industriji in športu.

Aleš Leonardis je profesor na University of Birmingham in sodirektor centra Centre for Computational Neuroscience and Cognitive Robotics na isti univerzi. Prav tako je profesor na Fakulteti za Računalništvo in informatiko Univerze v Ljubljani in pridruženi profesor na Fakulteti za računalništvo in informatiko Tehniške univerze v Gradcu. Njegovo raziskovalno področje obsega robustne in adaptivne metode računalniškega vida, kategorizacijo objektov in scen, statistične metode vizualnega učenja, modeliranje 3D objektov in biološko motivirani vid. Je član združenja IAPR ter IEEE in IEEE Computer Society.