

# Učenie reprezentácie objektov pomocou prekrývajúcich sa vývinových štruktúr

Andrej Lúčny – Michal Vician

Katedra aplikovanej informatiky, Fakulta matematiky, fyziky a informatiky, Univerzita Komenského  
Mlynská Dolina, 842 48 Bratislava  
lucny@fmph.uniba.sk, Michal.Vician@st.fmph.uniba.sk

## Abstrakt

Prezentujeme systém, ktorý rozpoznáva nové objekty v scéne pozorovanej kamerou. Objekty sú reprezentované množinou šablón význačných orientácií (rozpoznávanie na základe tvaru). Pokiaľ disponujeme vhodnou množinou šablón, sme schopní reprezentovaný objekt rozpoznať. Problém je, že táto reprezentácia sa obvykle získava pomocou učiteľa. Naš systém je schopný postupne vytvárať túto reprezentáciu na základe sledovania pohybu objektov v scéne detektorom pohybu (pohybujeme s nimi pomocou nití). Takto pohyb objektov umožní systému porozumieť, že daný objekt tvorí jeden celok a získa schopnosť ho rozpoznať na základe jeho tvaru. Spolupráca medzi detekciou pohybu a rozpoznávaním tvaru je inšpirovaná Minského myšlienkou prekrývajúcich sa vývinových štruktúr (v rámci societného modelu mysle).

## 1 Úvod

Marvin Minsky vo svojej legendárnej knihe *The Society of Mind* [9] uviedol množstvo inšpiratívnych myšlienok, ktoré sa neskôr stali predmetom pokusov o ich implementáciu. Jednou z nich je idea vývinu pomocou prekrývajúcich sa úrovní (*development by overlapping stages*). Našou snahou bolo demonštrovať ju na konkrétnom príklade. Vybrali sme si na to úlohu z počítačového videnia, konkrétne rozpoznanie objektov v scéne. V posledných rokoch bolo vyvinutých viac metód na reprezentáciu objektov nepravidelného tvaru. Najjednoduchšou a zároveň aj najrýchlejšou metódou je metóda šablón význačných orientácií (*dominant orientation templates*) – DOT [3]. Všetky tieto metódy vyžadujú fázu učenia, počas ktorej je vytvorená reprezentácia rozpoznávaných objektov a prebieha za špeciálnych podmienok: v prípade DOT je objekt vystavený voči kontrastnému pozadiu, nakoľko je založená na detekciu hrán objektu, pričom je objekt reprezentovaný sadou šablón zodpovedajúcim rôznym pohľadom. Človek však takto objekty nerozpoznáva: reprezentáciu objektu si vytvorí za štandardných podmienok. Musí teda nejako pochopiť, že sa díva na

niečo, čo tvorí jeden kus. Zobrali sme preto jednu (a pre jednoduchosť aj jediná) z možných metód, ako také niečo môže urobiť a to detekciu pohybu rozdielovou snímku [2]. Kamerou pozorujeme scénu, v ktorej budú na nitiach zavesené neznáme objekty. Pokiaľ sa niektorý z nich pohne, nižšia úroveň systému rozpozna hranicu pohybujúceho sa objektu. Tá približne (teoreticky presne, ale prakticky len približne) zodpovedá jeho skutočnej hranici, z čoho môže byť vytvorená jedna šablóna, zodpovedajúca jednému pohľadu. Môže sa stať, že k reprezentácii objektu neprináša nič nové, lebo takú šablónu už v sade pre daný objekt máme. V opačnom prípade reprezentáciu objektu obohatíme o ďalšiu šablónu. Tak sa stane, že po čase reprezentácia objektu dozreje a bude dostatočná, aby sme ho dokázali rozpoznať pomocou metódy DOT, predstavujúcej vyššiu úroveň. Je pritom teda potrebné kombinovať obe úrovne riadenia: z úvodu je významnejšia nižšia úroveň založená na detekcii pohybu, neskôr však nastane stav, keď dohrala svoju úlohu a rozpoznávanie sa opiera výlučne o vyššiu úroveň založenú na rozpoznávaní tvaru.

V kapitole 2 popíšeme bližšie Minského ideu, v kapitole 3 metódu detekcie pohybu, v kapitole 4 metódu rozpoznávania tvaru. V kapitole 5 popíšeme spôsob integrácie, ktorý nie je technicky triviálny a preto ho zakladáme na architektúre Agent-Space[5], vytvorenej práve pre implementáciu častí Minského societného modelu, vychádzajúc z princípov multi-agentových systémov [4].

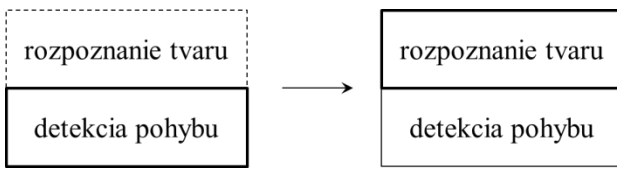
## 2 Vývin prekrývajúcich sa úrovní

Ide o predstavu, že systém nadobúda svoje schopnosti postupným zrením úrovní riadiacich štruktúr, pri ktorom vyššie úrovne disponujú lepšími schopnosťami než nižšie, avšak nie je možné ich naštartovať z ničoho. Nezaobídu sa bez nižších úrovní, ktorých schopnosti sú slabšie a ich úloha v systéme spočíva v tom, aby priviedli vyššie úrovne k funkčnosti. Istý čas, teda pracuje slabšie. Akonáhle však jeho určitá úroveň dospeje k istej saturácii schopností, ktoré si postupne buduje na základe dát vznikajúcich v nižších úrovniach, prevezme riadenie. Na systéme vtedy možno pozorovať kvalitatívny skok v jeho

schopnostiach. Systém je teda vopred štruktúrne usporiadaný tak, aby vývin jeho schopností postupne prechádzal rôznymi kvalitatívnymi fázami. Táto idea vychádza z vývinovej psychológie, hlavne z výskumu J. Piageta a snaží sa vysvetliť, prečo u detí dochádza k dramatickým a sľaby načasovaným, zlomom v ich schopnostiach.

V našom prípade budeme mať len dve vrstvy. Nižšiu, ktorá dokáže fungovať z ničoho, ale vidí v scéne len pohybujúce sa predmety a vyššiu, ktorá dokáže fungovať až po vytvorení dostatočnej reprezentácie objektov v scéne, ale vidí v scéne i nepohybujúce sa predmety, pokiaľ sú celé viditeľné. Ďalšie vrstvy by mohli viesť vyriešiť zmenu škály (zväčšenie, zmenšenie), zákryt a podobne, toto však presahuje náš aktuálny zámer.

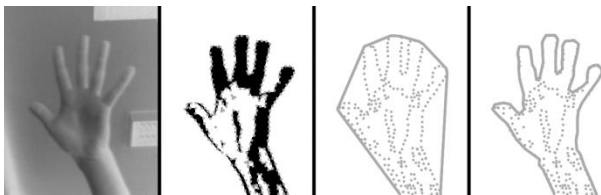
Nižšia vrstva funguje stále, ale dôležitá je len do okamihu, kedy si vyššia vrstva s jej pomocou vybuduje dostatočnú reprezentáciu rozpoznávaného objektu. Vtedy vyššia vrstva prevezme riadenie a schopnosti systému vykážu prudkú kvalitatívnu zmenu: predtým systém vidí len pohybujúce sa objekty, potom začne odrazu vidieť aj stojace (obr. 1).



Obr. 1. Vývin rozpoznávania na báze prekrývajúcich sa úrovní riadenia.

### 3 Detekcia pohybu.

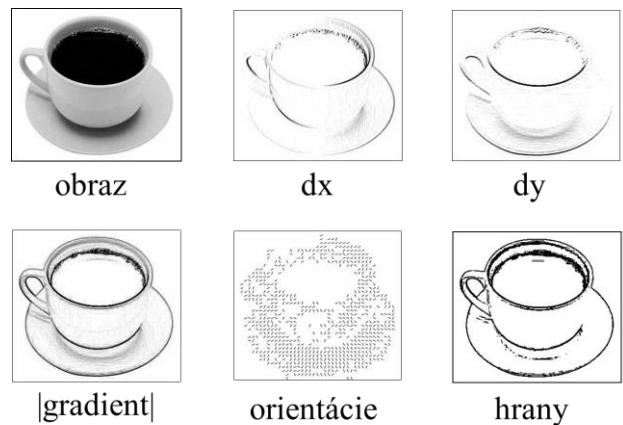
Nižšia úroveň systému rozpoznáva objekty na základe ich pohybu. Keďže kamera je v našom prípade statická, veľmi ľahko sa urobí rozdielový snímok. Ten poskytne množinu bodov obrazu, na ktorých dochádza ku zmene. Ako vidno na obr. 2, táto množina nedáva priamo hranicu objektu. Tú musíme spočítať ako najpravdepodobnejší nekonvexný ale celistvý tvar  $\chi$ -algoritmom [10] (obr. 2).



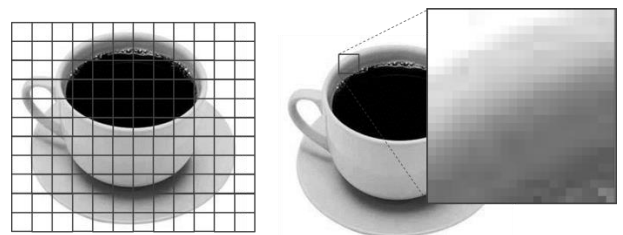
Obr. 2. Detekcia pohybu. Zľava doprava: obraz, rozdielový snímok, vstup do  $\chi$ -algoritmu, výstup z  $\chi$ -algoritmu

### 4 Rozpoznávanie tvaru.

Rozpoznávanie tvaru predstavuje v našom systéme vyššiu úroveň. Realizovali sme ju pomocou metódy DOT – metódy šablón význačných orientácií [3]. Táto metóda používa na reprezentáciu objektu sadu šablón, predstavujúcich jeho tvar z rôznych pohľadov. Každá šablóna vychádza z výsledkov hranového detektora (napr. Canny [2]), ktorý intenzity šedej spracuje Sobelovým operátorom [2] (obr. 3). Ten dáva odhad derivácie obrazu, t.j. vektor derivácií vo vodorovnom a zvislom smere (dx,dy) zvaný gradient. Keď zanedbáme smer tohto vektora, teda berieme tú z možností (-dx,-dy) a (dx,dy), ktorá má kladnú zložku y a určíme jej argument, dostávame v každom bode orientáciu – uhol od 0 po  $\pi$ . Budú nás pritom zaujímať len tie body obrazu, ktoré zodpovedajú hranám (získaných prahovaním veľkosti gradientu a následným stenčením). Ďalej sa obmedzíme na niekoľko málo (napr. 8) kvalitatívnych hodnôt orientácie, napr.  $0 \approx (0, \pi/8), 1 \approx (\pi/8, \pi/4), \dots$



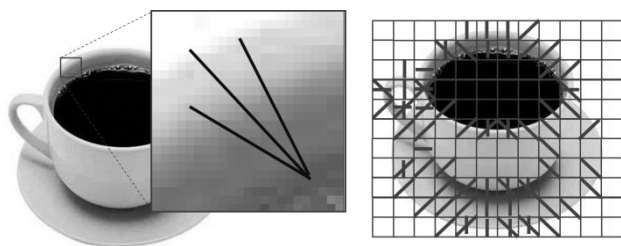
Obr. 3. Získanie hrán a orientácii gradientu obrazu.



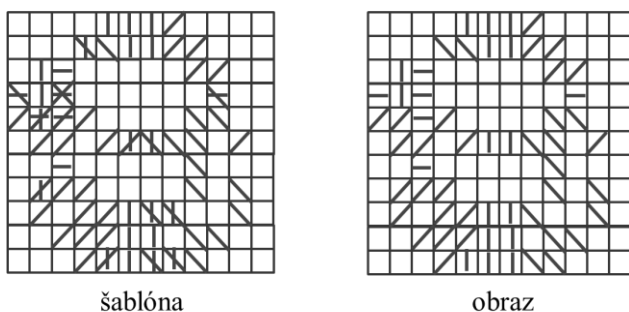
Obr. 4. Pokrytie obrazu regiónmi

Samotnú šablónu získame tak, že pokryjeme obraz neprekrývajúcimi sa regiónmi (obr. 4. vľavo). Ku každému regiónu potom vezmeme v úvahu

orientácie bodov patriacich k hranám (obr. 4. vpravo). Z týchto orientácií určíme sadu prevládajúcich (obr. 5).



Obr. 5. Určenie význačných orientácií



šablóna

obraz

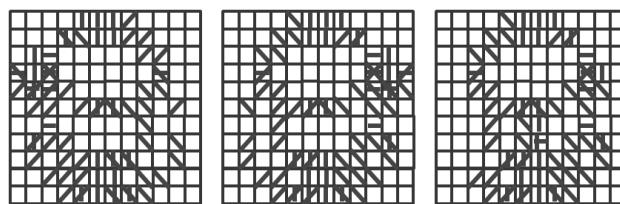
Obr. 6. Šablóna (vľavo) a prirovnávaný obraz (vpravo)

Tak dostaneme jednu šablónu. Tú potom môžeme prirovnáť k obrazu tak, že aktuálny obraz spracujeme analogickým spôsobom ako vzorový obraz pri tvorbe šablóny, ale v tomto prípade vyberáme len jedinú význačnú orientáciu z regiónu (obr. 6). Samotné porovnanie obrazu so šablónou potom prebieha testovaním, či pre drtivú väčšinu regiónov nájdeme v obraze jednu z orientácií šablóny. Šablónu musíme samozrejme prikladať na rôzne miesta obrazu. Bolo by však príliš náročné prikladať ju na každý pixel. Je výhodnejšie držať sa jedného položenia regiónov na obraz. V takom prípade ale zhoda nemusí nastať vôbec, pokiaľ sú regióny šablóny voči regiónom na obraze dostatočne posunuté. Môžeme si však narobiť z jedného vzoru viac šablón tak, že po ňom posúvame mriežku určujúcu regióny (obr. 7). V praxi nám to zďaleka nezvýši počet šablón o počet posunutí mriežky, lebo šablóny vychádzajú často rovnaké. Pri miernej strate presnosti (DOT je nepresná tak či tak) je možné dokonca zosumarizovať rôznym položením mriežky vygenerované šablóny do jedinej. Urobíme to tak, že v regióne šablóny budú všetky orientácie z okolitých poposúvaných regiónov. Takú šablónu nie je potrebné posúvať po pixloch, hoci už tak presne nevystihuje reprezentovaný objekt.



Obr. 7. Šablóny z rôznych posunutí regiónov

Z jedného pohľadu na objekt teda dostaneme jednu šablónu. Z rôznych pohľadov dostávame celú sadu šablón. Zdanlivo je ich opäť príliš moc. Keďže však nevyžadujeme 100% zhodu šablóny s obrazom, môžeme si dovoliť jednou šablónou nahradiť všetky jej podobné, v rámci podobnosti, ktorú pripúšťame pri porovnaní. Na väčšinu objektov nám stačí menej šablón, než by sme očakávali (obr. 8).



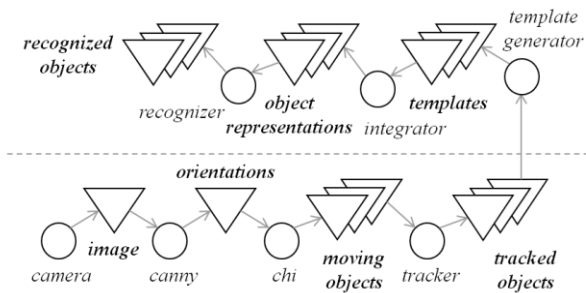
Obr. 8. Reprezentácia objektu

Porovnanie šablón sa dá implementovať veľmi efektívne pomocou bitových operácií [3]. DOT je preto veľmi rýchla metóda, použiteľná v reálnom čase. S kvalitnou kamerou dáva dobré experimentálne výsledky, ktoré sa vyrovnávajú i oveľa pomalším metódam, ako je napr. HOG [10].

## 5 Integrácia úrovni cez Agent-Space

Detekcia pohybu produkuje z každého snímku kamery sadu objektov, ktoré sa pohli. Ich pozícia sa porovnáva voči predchádzajúcej snímke, čo zodpovedá sledovaniu pohybu objektov na obraze. Vďaka tomu vieme, ktoré generované šablóny k sebe patria a tvoria reprezentáciu jedného objektu. Ich akumuláciou (samozrejme úspornou v zmysle metódy DOT) vzniká postupne reprezentácia objektu. Výrazná podobnosť medzi dvomi vzniknutými reprezentáciami vedie k zlučeniu dvoch reprezentácií, keďže zrejme ide o rovnaký objekt. Keď zistíme, že detekcia pohybu síce generuje dost' ďalších šablón, ale tieto už neobohacujú reprezentáciu, spustíme detekciu objektu na základe tvaru.

V systéme prebieha súčasne viacero procesov, ktoré majú navyše rôznu rýchlosť. Skoordinovať ich preto nie je priamočiare. My sme na tento účel použili vlastnú architektúru Agent-Space, [www.agentspace.org](http://www.agentspace.org) [3], pomocou ktorej sme implementovali jednotlivé procesy v podobe reaktívnych agentov komunikujúcich nepriamo, cez tzv. *space* (obr. 9).



Obr. 9. Schéma systému v Agent-Space

## 6 Záver

Snažili sme sa ukázať jednoduchý ale reálny príklad na Minského ideu vývinu pomocou prekrývajúcich sa úrovní riadenia, ktorá sa odráža v zlomových zmenách v schopnostiach vyvíjajúceho sa systému. Systém, ktorý sme ako príklad predložili, kombinuje dve metódy rozpoznávania: na základe pohybu a na základe tvaru. Kým prvá metóda je schopná pracovať sama o sebe, druhá musí mať najprv vytvorenú dostatočne bohatú reprezentáciu. To sa udeje na základe výstupu z prvej metódy. Systém tak najprv vidí len pohybujúce sa objekty a potom zlomovo získava schopnosť vidieť aj objekty nepohybujúce sa.

Na implementáciu sme použili C++, pthread a OpenCV. Podstatnú časť implementácie systému vytvoril jeden z autorov v rámci svojej diplomovej práce [11]. K práci je možné pozrieť si aj demonštračné video <http://youtu.be/xpdvMPGmTnc>

Ďalšie možnosti rozvoja tohto projektu by spočívali v integrácii ďalších metód rozpoznávania objektov za účelom získania schopností, ktoré tento systém momentálne nemá. V prvom rade tento systém nepodporuje žiadny z invariantov okrem posunutia. Otočenie zvláda rozšírením sady šablón. Absolútne nezvláda zväčšenie a zmenšenie, čiže pri pohybe objektov nie je moc vhodné pohybovať s nimi pred kamerou dopredu a dozadu, radšej doľava a doprava a točiť s nimi. Nevie navyše riešiť zákryt objektov. Od rozpoznávania objektov, ktoré dokáže vykonať človek, má tento systém každopádne veľmi ďaleko.

## PodĎakovanie

Tento príspevok vznikol za podpory grantovej agentúry ASFEU v rámci grantovej úlohy KC-INTELSYS, ITMS 26240220072.

## Literatúra

- [1] R. A. Brooks: *Cambrian Intelligence*, The MIT Press, Cambridge, Mass., 1999.
- [2] Davies, E. R.: *Machine Vision: Theory, Algorithms, Practicalities*. Elsevier, 2005
- [3] Hinterstoisser, S. - Lepetit, V. - Ilic, S. - Fua, P. - Navab, N.: *Dominant Orientation Templates for Real-Time Detection of Texture-Less Objects*. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, California (USA), June 2010
- [4] Kelemen, J.: *The Agent Paradigm*. Computing and Informatics, Vol.22. (2003), pp. 513-519
- [5] Lúčny, A.: *Building Complex Systems with Agent-Space Architecture*. Computing and Informatics, Vol. 23 (2004), pp. 1001-1036
- [6] Lúčny, A.: *Od medzimodulových spojení k nepriamej komunikácii medzi agentami*. Znalosti, VŠE Ostrava, 2007.
- [7] Lúčny, A.: *Advantages of Multi-agent Approach to Building of Monitoring Systems*. In: Informatics 2007 (Ivan Plander ed.), SSAKI, Bratislava, 2007
- [8] Lúčny, A.: *Multiagentový prístup k modelovaniu mysle - alebo ako sledovať pingpongovú loptičku*. In: Umelá inteligencia a kognitívna veda III. (Kvasnička V. ed.), STU, Bratislava, 2011
- [9] Minsky, M.: *Society of Mind*. Simon & Schuster, New York, 1986
- [10] Sonka, M. – Hlaváč, V. – Boyle, R.: *Image processing, analysis and machine vision*. 3rd edition, Thomson Learning, Toronto, 2007
- [11] Vician, M.: *Formovanie riadiacich štruktúr prekrývaním úrovní vývinu*. Diplomová práca, FMFI UK Bratislava, 2011