

Robot pred zrkadlom

Andrej Lúčny

Katedra aplikovanej informatiky, Fakulta matematiky, fyziky a informatiky, Univerzita Komenského
Mlynská dolina, Bratislava
lucny@fmph.uniba.sk

Abstrakt

Prezentujeme riadiaci systém robota, ktorý interaguje so svojím obrazom v zrkadle za účelom pochopenia, že sa pozerá sám na seba. Skúmame pritom z akých jednoduchších mechanizmov schopnosť uvidieť sa v zrkadle môže povstať. Vychádza nám nasledovné. V prvom rade je to schopnosť vytvárať si model vlastného tela. V druhom rade schopnosť vytvárať analogický model z pozorovaného tela. Asociácia videného s vlastným telom potom môže vzniknúť na základe dokonalej korelácie medzi týmito modelmi. Potrebujeme však ešte nejaký mechanizmus, ktorý uvedie oba tieto modely do pohybu – dobre tu posluží napríklad imitácia. Dôležitá je ďalej kategorizácia takýchto imitácií, typická pre členov nejakej society. Schopnosti rozlíšiť seba od ostatných tu podľa nás prechádza schopnosť rozlíšiť ostatných navzájom.

1 Úvod

Pri pohľade do zrkadla vieme, že sa pozeráme sami na seba. Zrkadlá nás obklopujú od malička, ale aj keď sa zo zrkadlom prvý krát stretne autochtónny obyvateľ pralesa, pomerne rýchlo zistí, že sa v zrkadle pozerá na seba a to aj bez toho, že by mu niekto princíp zrkadla vysvetlil. Naopak deti mladšie než 18 mesiacov sa v zrkadle nevidia, ale napríklad hľadajú iné dieťa v priestore za zrkadlom (Obr. 1c) (Galup 1970).

1.1 Pohľad do zrkadla v prírode

Iba málo biologických druhov má rovnakú schopnosť. Nie je to vždy ľahké otestovať, avšak pomerne slušným dôkazom je úspech v tzv. dot-mirror teste, kedy sa najprv nechá testovaný subjekt so zrkadlom oboznámiť (aby vedel ako v zrkadle vyzerá) a potom sa skúma či na svojom tele hmatá po dodatočne (a nenápadne) urobenej bodke kriľavej farby na mieste, ktoré môže vidieť iba v zrkadle. Týmto testom úspešne prechádzajú všetky druhy ľudopopov, slony, delfíny, kosatky a straky. To nemusia byť všetky zvieratá, ktoré sa v zrkadle vidia, lebo napríklad krkavec je vo všeobecnosti považovaný za oveľa inteligentnejšieho ako straka, na rozdiel od nej však nemá záľubu v jagavých predmetoch, takže vykonať s ním dot-mirror

test je ťažšie. Na druhej strane je možné, že u straky dot-mirror test dokazuje len to, že vie uchopovať jagavé predmety aj podľa obrazu v zrkadle a pritom nechápe, že sú umiestnené na jej tele. Dot-mirror test taktiež požaduje o niečo viac než samotné uvidenie sa (pod čím rozumieme pochopenie, že sa dívame na seba) a to vytvorenie zapamätanie si svojho obrazu.



a)



b)



c)

Obr. 1: Šimpanz, mačka a dieťa pred zrkadlom (šimpanz sa predvádza, mačka útočne cerí zuby, dieťa mladšie ako 18 mesiacov hľadá iné dieťa za zrkadlom)

Takže, pokiaľ aj zvieru neprejde dot-mirror testom, možno v určitých prípadoch oprávnené usúdiť, že sa v zrkadle vidí alebo nie, podľa pocitu bezpečia alebo naopak podľa pocitu ohrozenia, ktorý dáva najavo spôsobom charakteristickým pre svoj druh. Kým šimpanz sa zjavne pred zrkadlom predvádza (Obr. 1a), mačka má tendenciu na obraz v zrkadle zaútočiť (Obr. 1b).

Na základe poznatkov z prírody teda vieme, že schopnosť uvidieť sa v zrkadle je naviazaná na postupný vývin kognitívnych schopností jedinca a povstáva v určitom veku. Je v prírode zriedkavá a vyskytuje sa len u druhov, ktoré sa v istých ohľadoch ponášajú na človeka (ak vynecháme straky, tak všetky žijú sociálnym spôsobom života, majú rozvinutú schopnosť vnímania vlastného tela a majú veľký pomer medzi veľkosťou mozgu a tela).

1.2 Umelý pohľad do zrkadla

V tomto článku sa venujeme otázke z akých jednoduchších mechanizmov či modulov je možné schopnosť uvidieť sa v zrkadle poskladať. Nápadom sú jednak uvedené poznatky z prírody, jednak svoje predstavy budeme testovať na umelom systéme, na ktorom sa budeme snažiť uvidenie sa v zrkadle demonštrovať.

Zďaleka nie sme prví, kto sa o niečo také pokúšal. Vychádzame hlavne z práce Scasellatti a Hart 2012 (Scasellatti bol svojho času jeden z tvorcov robota COG v AiLab na MIT), podľa ktorých sa robot (alebo človek či zvieru) v zrkadle vidí na základe dokonalej synchronizácie medzi pohybom tela a pohybom jeho obrazu v zrkadle. Ich prístup je pomerne náročný, lebo si dali za cieľ zostrojiť robota (Nico), ktorý dokáže konať rôzne akcie podľa obrazu v zrkadle a to v priestore, ktorý inak nevidí.

Ich prístup považujeme za oveľa vhodnejší, než autorov robota QBO (Corpora 2011), ktorí vyhlasujú, že ich robot prechádza dot-mirror testom. Robí tak však na základe informácie od učiteľa, ktorý mu povie: "Teraz vidíš seba". Takže keby stretol svojho dvojníka, tak povie: "Toto som ja", zatiaľ čo Nico by vedel, že to nemôže byť on, lebo hoci vyzerá rovnako, hýbe sa ináč.

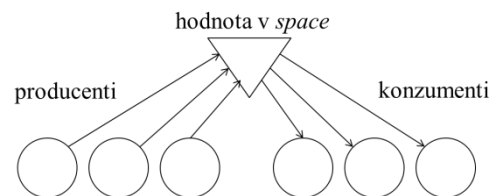
Na druhej strane z technického hľadiska by bolo pre nás príliš náročné urobiť robota ako je Nico a hlavne vyhodnocovať charakter pohybov tela pri autentickom modeli tela. Na skúmanie komponentov, ktoré za uvidením sa v zrkadle stoja, to našťastie nie je nevyhnutné, ako svojim prístupom ukázal Takeno 2008, pracujúci s veľmi jednoduchým robotom a jednoduchým modelom tela. My zvolíme strednú cestu, bližšiu skôr prístupu Takena: budeme pracovať s humanoidným robotom, avšak jeho pohyby výrazne obmedzíme, čím si zjednodušíme model jeho tela.

Konkrétne budeme používať simulátor humanoidného robota iCub, zvaný iCubSim (Sandini a kol. 2007) (Obr. 2). Tento simulátor je pomerne kvalitný a poskytuje nielen ovládanie 53-och stupňov

voľnosti, ale aj ich monitorovanie, t.j. model tela. Je v ňom teda potenciálne rozdiel medzi pokynom, akú polohu má určitý kĺb zaujať a skutočnosťou, ktorá (ako-tak) rešpektuje zákony fyziky (používa fyzikálny engine ODE).



Obr. 2: Simulátor iCubSim



Obr. 3: Dátový tok v architektúre Agent-Space. (kruh označuje modul, trojuholník hodnotu v *space*)

1.3 Architektúra

Využívať budeme taktiež výsledky vlastnej predchádzajúcej práce v oblasti tvorby modulárneho softvéru, konkrétne architektúru Agent-Space (Lúčny 2004). Táto nám umožní realizovať jednotlivé mechanizmy uvidenia sa ako na sebe nezávislé moduly pracujúce paralelne a vzájomne interagujúce prostredníctvom tzv. čiernej tabuľe (zvanej *space*). Tu nám dosť pomôže, že pri charaktere tejto architektúry zaniká potreba zaoberať sa faktom, že jednotlivé moduly majú rôznu rýchlosť spracovania údajov. Základnou myšlienkou toho, ako je to dosiahnuté, je, že kým jeden modul pravidelne prepočítava svoj výsledok a zapisuje ho na tabuľu, druhý ho odtiaľ pravidelne číta a nestaráme sa pritom, či nejaký výsledok prečítať nestihne. Podobne nevedí, že sa spracúvajú výsledky vypočítané viacerými modulmi v rôznom čase, hlavné je, že berie tie najnovšie. Jeden výsledok môžu pritom počítať viaceré moduly, pričom jeho odberatelia nevedia, od koho pochádza. No a pritom jeden producent výsledku môže mať (ale nemusí) vyššiu prioritu než druhý, takže výsledok od druhého sa berie v úvahu len vtedy, keď nie je k dispozícii výsledok od prvého. Každý výsledok môže mať (a spravidla aj má) ohraničenú časovú platnosť, takže výsledok od prioritnejšieho producenta po čase zanikne a uvoľní

priestor pre výsledok od menej prioritného producenta, pokiaľ ten prioritnejší nevyprodukuje výsledok ďalší (Obr. 3).

Táto architektúra je silne založená na myšlienkach subsumpčnej architektúry (Books 1999) a societného modelu mysle (Minsky 1986), avšak tieto myšlienky sú vyjadrené v jazyku multi-agentových systémov (Kelemen 2001).



Obr. 4: Simulátor iCubSim rozšírený o kameru

2 Implementačné prostriedky

Naším cieľom je poodhaliť podstatu procesu uvidenia sa v zrkadle pomocou implementovania umelého systému, v ktorom tento proces prebieha. Budujeme ho z modulov reprezentujúcich mechanizmy podieľajúce sa na tomto procese, takže skúmame z akej štruktúry jednoduchších procesov tento proces povstáva.

V simulátore iCubSim simulovaný robot vidí virtuálnu scénu, ktorej je súčasťou, napríklad stôl s predmetmi. My ho integrujeme do systému, kde je rozšírený o fyzickú kameru, snímajúcu priestor pred monitorom, na ktorom je renderovaná postava iCub-u (Obr. 4). Vďaka tomu bude simulovaný iCub miesto svojej virtuálnej scény vidieť priestor pred monitorom, v ktorom sa nachádza programátor, t.j. človek. Pre naše účely je nevyhnutné, aby boli modely tela človeka a robota vzájomne porovnateľné – čiže nám vyhovuje, že iCub je humanoidný robot.

Ako sme však už spomenuli, model tela výrazne zjednodušíme, a to na jediné číslo – uhol náklonu hlavy na stranu (rovno 0°, doľava zhruba do -55°, doprava zhruba do +55°). S ostatnými kĺbmi iCub-u nebudeme hýbať a pohľad do virtuálnej scény s iCub-om zvolíme tak, aby na obrazovke monitora bola renderovaná len hlava.

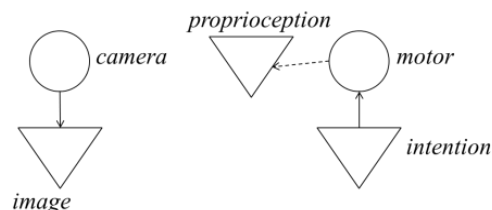
3 Mechanizmy

V zmysle zamýšľanej stratégie uvidenia sa v zrkadle, je našim cieľom implementácia systému, v ktorom by

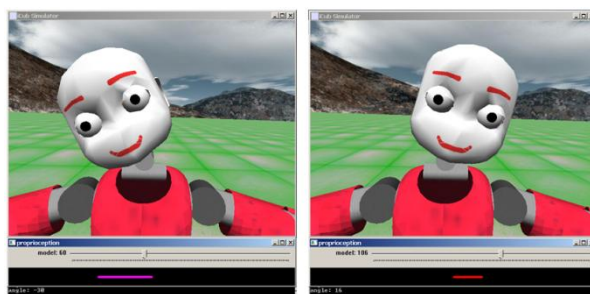
bolo možné vyhodnotiť koreláciu medzi modelom tela, ktoré systém má a analogickým modelom tela, ktoré systém vidí. Myslíme pritom na biologickú relevantnosť komponentov, z ktorých systém skladáme, majúci na mysli špecifické vlastnosti tých biologických druhov, ktoré sa v zrkadle vidia. Systém budujeme inkrementálne, opierajúc sa o zvolenú modularitu. Pridaním ďalších modulov implementuje vždy ďalší mechanizmus, o ktorom predpokladáme, že za uvidením sa v zrkadle stojí.

3.1 Model tela (propriocepcia)

V prvom rade musíme mať k dispozícii oba porovnávané modely. Začneme s modelom vlastného tela robota. Ten už máme implementovaný v simulátore, takže ho musíme akurát prispôsobiť našej architektúre, t.j. dostať ho na onu čiernu tabuľu (do *space*). Implementujeme preto modul "motor", ktorý zo *space* číta želanú hodnotu nastavenia tela "intention", komunikáciou so simulátorom (prostredníctvom yarp rpc protokolu) sa ju snaží nastaviť telu robota, monitoruje skutočné nastavenie, ktoré sa podarilo dosiahnuť a zapisuje ho do *space* ako hodnotu "proprioception" (Obr. 5). Vďaka tomu vieme zápisom do *space* ovládať telo robota a čítaním zo *space* monitorovať skutočný stav jeho tela (Obr. 6).



Obr. 5: Implementačná fáza 1.



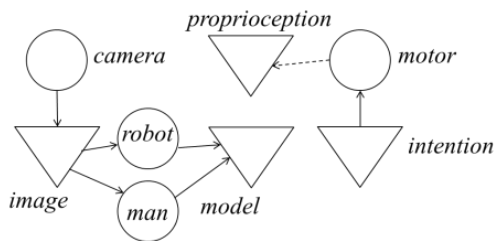
Obr. 6: Zjednodušený model tela robota (zobrazený ako vodorovná palička v čiernom poli dolu)

Paralelne k ovládaniu aktuátorov zabezpečujeme vstup zo senzorov. Naš jediný senzor je kamera, z ktorej obraz zapisujeme pravidelne do *space* (hodnota "image"). Moduly spracúvajúce tento obraz zďaleka nestihnú zo *space* prečítať každú snímku do *space* zapísanú, ale vďaka zvolenej architektúre sa tým nemusíme nijako špeciálne zaoberať.

3.2 Zrkadlenie

Teraz prejdime k získaniu analogického modelu tela, ktoré vidíme na obraze. Neriešime pritom, odkiaľ sa takáto netriviálna schopnosť (t.j. schopnosť popísať parameter svojho tela na podobnom tele videnom na obraze) berie, proste ju "nadrôtujeme".

Keďže v našom experimentovaní zamýšľame, aby sa robot pozeral na obraz človeka i na svoj obraz (t.j. obraz robota), z oboch týchto obrazov musíme vedieť používaný model tela vytvoriť. Žiaľ, hoci iCub je humanoidný, jeho obraz sa od obrazu človeka natoľko líši, že sa nám nepodarilo použiť na tento účel jednu spoločnú metódu. Avšak to nám v princípe neprekáča, lebo proces rozpoznávania si aj tak predstavujeme ako viacero nedokonalých metód zosúladených do funkčného celku. V našom prípade teda vyvineme dve metódy, jedna bude spracúvať obraz robota, druhá človeka. Obe budú vytvárať rovnaký model tela. Avšak nielen to: ich výstupy budú zapísané na *space* do rovnakej hodnoty (hodnota "model" na Obr. 7). Keby sa teda ozvali obidve naraz (čo je málo pravdepodobné), navzájom by si svoj výsledok prepisovali. Každopádne však pritom ostatné moduly nemajú prístup k informácii od ktorej metódy hodnota modelu pochádza.



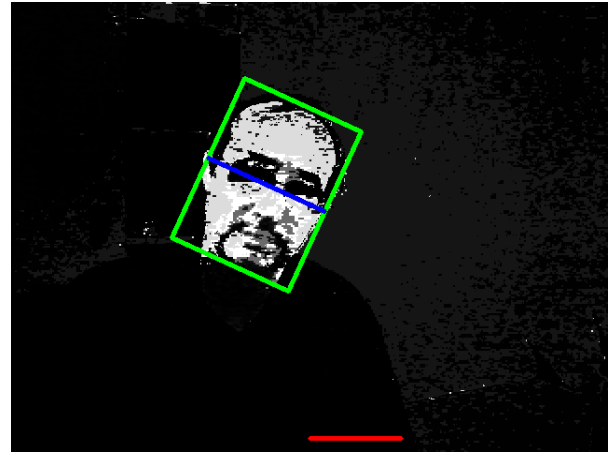
Obr. 7: Implementačná fáza 2



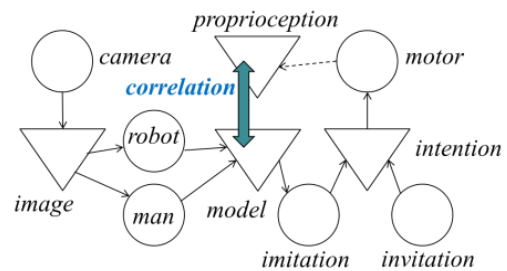
Obr. 8: Získanie modelu z obrazu robota (SURF)

Vytvorenie modelu z obrazu robota sme testovali tak, že sme pred kamerou otáčali vytlačenú fotku monitora s renderovaným obrazom robota. Najlepšie sa nám osvedčilo použiť metódu SURF (Bay a kol. 2008). Tá na ľubovoľnom obraze nájde tzv. význačné body a opíše ich okolie tzv. deskriptorom a to nezávisle na posunutí, rotácii a zväčšení či zmenšení. To umožňuje

SURFu spárovať význačné body s podobnými deskriptormi získané zo vzoru (obrázok tváre robota) a obrazu (aktuálny snímok z kamery). Z nich sa dá vypočítať projekcia vzoru na obraz a z tej už ľahko určíme uhol náklonu vzoru, čo je parameter zjednodušeného modelu tela, ktorý hľadáme (Obr. 8).



Obr. 9: Získanie modelu z obrazu človeka (Haar cascades a CamShift)



Obr. 10: Implementačná fáza 3

Kým jeden iCub je podobný druhému ako vajce vajcu, ľudia sú rôzni. Na rozpoznanie náklonu ich hlavy do strany preto použijeme inú metódu. Pokiaľ je hlava vzpriamene, tvár dokážeme spoľahlivo identifikovať pomocou face detektora. Používame hotový detektor z OpenCV, založený na Haarových príznakoch v kaskádach (Viola a Jones 2001). Ide o klasifikátor naučený z pozitívnych a negatívnych prípadov. Jednotlivé príznaky si možno predstaviť ako binárnu masku malej veľkosti položenú na konkrétne miesto na obraz vzorov. Kaskády zase ako rozhodovací strom budovaný na základe čo najlepšej rozlišovacej schopnosti príznakov. Tento detektor tváre však funguje len vo vzpriamenej polohe. Pri náklone nám teda nepomôže. Poskytne nám však objekt na obraze, ktorý môžeme potom sledovať pomocou bežných metód (tracking). Používame metódu CamShift (Bradski 1998), ktorá je založená na histograme farieb sledovanej oblasti. Iteratívne sa potom vyhodnocujú body na aktuálnom obraze farebne najpodobnejšie iniciálnej oblasti. Sledovaná oblasť sa potom iteratívne posúva a rotuje podľa ťažiska oných dostatočne

podobných bodov na nových snímkach. Z rotácie sledovanej oblasti potom priamo dostávame potrebný model. (Obr. 9). Všetky použité algoritmy sú implementované v OpenCV (www.opencv.org) (Bradski 2000).

```
class imitationAgent : public Agent
{
private:
    string modelInclinationBlock;
    string inclinationBlock;
protected:
    void init (string args);
    void sense_select_act (int pid);
public:
    imitationAgent (string args) : Agent(args) {};
};

void imitationAgent::init (string args)
{
    ifstream in(args);
    in >> modelInclinationBlock;
    in >> inclinationBlock;
    timer_attach(1000,1000);
}

void imitationAgent::sense_select_act (int pid)
{
    double undefined = 360.0;
    double inclination =
        space_read(modelInclinationBlock, undefined);
    if (inclination != undefined) {
        space_write(inclinationBlock,inclination,1500);
    }
}
```

Obr. 11: Ukážka kódu modulu realizujúceho imitáciu. Druhý argument `space_read` je preddefinovaná hodnota pre prípad, že hodnota v *space* nie je definovaná. Tretí argument `space_write` je časová platnosť v ms. (Kód je zjednodušený.)

3.3 Imitácia

Disponujeme už modelom svojho tela (hodnota "intention") aj modelom videného tela (hodnota "model"). Koreláciu medzi týmito dvomi modelmi je však možné vyhodnotiť len pokiaľ sa tieto hodnoty v čase menia. Potrebujeme preto uviesť obe telá do pohybu. Jedným z mechanizmov ako to zariadiť je imitácia.

Imitáciu môžeme veľmi ľahko implementovať, nakoľko oba modely sú kompatibilné. Je preto možné priamo kopírovať hodnotu "model" do hodnoty "intention" (viď Obr. 10). Keď potom robot uvidí obraz človeka alebo robota, dá hlavu do rovnakej polohy ako majú oni. Tento modul (voláme ho "imitation") je tak jednoduchý, že jeho kód môžeme poľahky uviesť (Obr. 11). Dôležité je všimnúť si, že hodnota "model" má obmedzenú časovú platnosť. Takže keď človek či robot zmizne z obrazu, po vypršaní tejto platnosti, hodnota zo *space* zmizne a v dôsledku toho sa hlava robota vráti

do vzpriamenej polohy, keďže práve to je preddefinovaná hodnota pre prípad, že v *space* nie je "intention" definovaná.

Takže teraz keď sa človek posadí pred kameru, iCub imituje jeho pohyby hlavy (Obr. 12). Pokiaľ chceme aby aj robot človeka nejako vyzýval k imitácii, môžeme doplniť modul "invitation", ktorý v prípade, že vidí človeka, ktorého model sa moc nemení, čas od času vygeneruje vhodný pohyb hlavy robota. To spraví tak, že náhodne vygeneruje uhol a zapíše ho do *space* – do hodnoty "intention" – pričom sa mu zide vyššia priorita zápisu, než má modul "imitation".

Nič menej sa ukázalo, že na predvedenie uvidenia sa v zrkadle tento modul nie je nevyhnutný. Je to vďaka tomu, že určovanie modelu videného tela má len určitú presnosť, takže jeho hodnoty prirodzene kolíšu aj keď sa videné telo vôbec nepohne. Tieto drobné odchýlky stačia na to, aby vyvolali imitačný proces, keď sa robot díva na svoj obraz v zrkadle.

3.4 Sociálne modelovanie

Imitácia nám poskytuje dostatočné dáta na vyhodnocovanie korelácie medzi modelom tela a modelom videného (viď obojsmerná šípka na Obr. 10). Na to, aby sa tvor pozerajúci sa do zrkadla uvidel, musí nejako tieto hodnoty kategorizovať. Je pomerne ľahké si predstaviť, že pre tvora žijúceho v nejakej societe má význam vyhodnocovať, ktorý z jedincov society je viac a ktorý menej ochotný imitovať, takže s obrazom jedinca spája aj informáciu o miere korelácie medzi uvažovanými modelmi. Primárne teda rozpoznáva členov svojej society. V zrkadle potom uvidí ako keby ďalšieho člena society, ktorý má nenormálne vysokú mieru onej korelácie, t.j. absolútnu ochotu imitovať. A to je on sám.

Pre naše účely sme túto kategorizáciu neimplementovali, ale zamerali sme sa na to, či v zázname dát vieme rozlíšiť situáciu, kedy sa robot díva na človeka a kedy na svoj obraz v zrkadle.

4 Výsledky

Postupne sme vyvinuli systém, do ktorého sme vložili všetky moduly, o ktorých sa domnievame, že hrajú určitú rolu v procese uvidenia sa v zrkadle. Keď teraz imitujúceho človeka pred kamerou nahradíme zrkadlom, robot imituje vlastný obraz v zrkadle (Obr. 13). Z úvodu sa chvíľu nič nedeje, ale onedlho nepresnosť v určení modelu videného tela spôsobí, že v "myslí" robota sa obraz mierne pohne a on sa to podujme imitovať. Následne sa rozbehne imitačný proces, však oveľa pomalšie, sústredenejšie a vo výraznejšej miere (náklon hlavy bude väčší než obyčajne), než je tomu pri interakcii s človekom.

Pritom matematicky rozlíšiť dáta nie je jednoduchá záležitosť a s učebnicovou definíciou korelácie tu nevystačíme. Nič menej, pohľad na priebeh

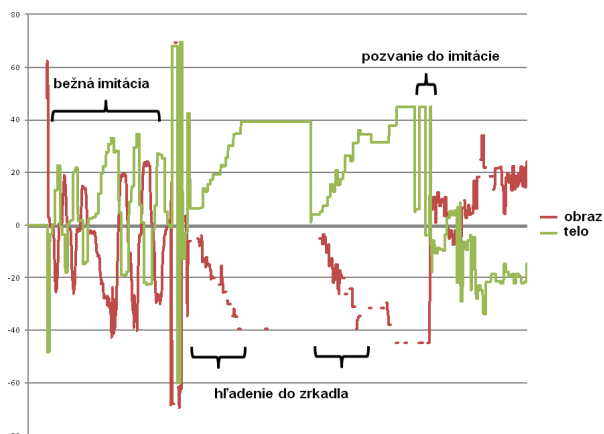
dát je veľavravný a poľahky z nich určíme, kedy sa robot díval do zrkadla (Obr. 14).



Obr. 12: Imitácia



Obr. 13: Robot sa vidí v zrkadle



Obr. 14: Záznam modelov viditeľného a vlastného tela (os x predstavuje čas, os y uhol v stupňoch) (úplnejšia čiara je model vlastného tela, miestami prerušovaná čiara je model viditeľného tela – vlastné telo máme vždy, ale nie vždy vidíme nejaké telo a aj keď nejaké vidíme, nie vždy to rozpoznáme)

5 Záver

V tomto článku sme predviedli demonštračný systém v ktorom prebieha uvidenie sa v zrkadle, pričom povstáva z jednoduchších mechanizmov. Uvidenie sa prebehlo na základe dokonalej korelácie medzi vlastným telom a telom robota, nevyskytujúcej sa pri uvidení tela niekoho iného. Dokonca badať istú podobnosť medzi vyprodukovaným pohybmi hlavy robota pred zrkadlom a pohybmi šimpanza pred zrkadlom.

Nie všetko máme v tejto chvíli dotiahnuté do dokonalosti, nemáme napríklad vykonaný experiment, kde proti sebe postavíme dva simulátory iCubSim, ani nám nie je celkom jasné čo by v takom experimente malo vyjsť. Lebo je to situácia obdobná tomu, keď šikovný herec v prítomnosti imituje zrkadlo. Také experimenty sa robili a podarilo sa viacero ľudí oklamať, že sa dívali na svoj obraz. Tu by práve dosiaľ v podstate nepotrebný modul "invitation" mohol zohrať rozhodujúcu rolu.

Každopádne môžeme vysloviť ako hypotézu, že schopnosti uvidieť sa v zrkadle predchádza v prírode schopnosť uvidieť niekoho iného (a spoznať ho ako individuum). A taktiež, že dobrým predpokladom uvidenia sa je schopnosť konštruovať z viditeľného model analogický vlastnému telu, čo je schopnosť prekvapujúca, ale zrejme v prírode prítomná (vid' zrkadliace neuróny). Iste by bolo veľmi zaujímavé skúmať, kde táto schopnosť berie a ak z niečoho povstáva, prečo nie je úplne bežná. Ako aj to, nakoľko je vrodená – možno sme takým druhom, ktorý mal práve to šťastie, že sa jeho zrkadlová kôra vyvinula tak, že vyprodukovala útvar podobný motorickej kôre... Ťažko povedať...

Uvidenie sa v zrkadle je kognitívna funkcia, ktorú – úmerne jej zriedkavosti v prírode – vnímame ako niečo úžasné. V prvej chvíli sme preto nadšení, že ju vieme vyvolať u robota. Avšak v druhej chvíli si uvedomíme že sme ju zložili z malého počtu modulov a to modulov dosť jednoduchých. A ten úžas sa stráti.

Na záver pripomeňme, že uvidenie sa v zrkadle nie je ešte pochopením vlastnej existencie. My sme robili robota, ktorý vie, že: "Toto som ja", nie že "Ja som". Aj schopnosť prejsť dot-mirror testom sa u detí prejavuje vo veku 18 mesiacov, zatiaľ čo ako-tak rozvinutú osobnosť má dieťa zhruba až vo veku dvoch rokov.

Prezentačné video je dostupné na adrese:

www.agentspace.org/mirror/iCubSimAtTheMirror.mp4

Zdrojové kódy sú dostupné na adrese:

www.agentspace.org/mirror/iCubSimAtTheMirror.zip

Literatúra

Bay, H. – Ess, A. – Tuytelaars, T. – Van Gool, L. (2008) Speeded-Up Robust Features (SURF).

Computer Vision and Image Understanding archive.
Volume 110 Issue 3, June, 2008, pp. 346-359

- Bradski, G. (1998) Computer vision face tracking for use in a perceptual user interface.
- Bradski, G. (2000) The OpenCV Library. *Dr Dobb's Journal of Software Tools*.
- Brooks, R. (1999). *Cambrian Intelligence*. The MIT Press, Cambridge, MA
- Burian Jan (2012) Konceptuální model prostředí pro seberozpznávajícího robota (The conceptual model of environment for self-recognition robot). <http://docplayer.cz/4089023-Konceptualni-model-prostredi-pro-seberozpznavajiciho-robota.html>
- Gallup, G. G., Jr. (1970). Chimpanzees: Self Recognition. *Science*, Vol. 167, 86-87.
- Hart, J. W. Scassellati, B. (2012). Mirror Perspective-Taking with a Humanoid Robot. In: *Proceedings of the 26th AAAI Conference on Artificial Intelligence (AAAI-12)*. Toronto, Canada.
- Kelemen, J. (2001). From statistics to emergence – exercises in systems modularity In: *Multi-Agent Systems and Applications*, (Luck, M., Maøík, V., Štepánková, O. Trappl, R.), Springer, Berlin, pp. 281-300
- Lucny, A. (2004). Building complex systems with Agent-Space architecture. *Computing and Informatics*, Vol. 23, pp. 1001-1036
- Minsky, M. (1986). *The Society of Mind*. Simon&Schuster, New York
- Sandini, G. – Metta, G. – Vernon, D. (2007). The iCub cognitive humanoid robot: an open-system research platform for enactive cognition. In: *50 years of artificial intelligence*, pp. 358-369, Springer-Verlag, Berlin
- Takeno, J. (2008). A Robot Succeeds in 100% Mirror Image Cognition. *International Journal on Smart Sensing and Intelligent Systems*, Vol. 1, No. 4, December 2008.
13. Viola, P. – Jones, M. (2001) Rapid object detection using a boosted cascade of simple features. *Conference on Computer Vision and Pattern Recognition*.
14. The Corpora. (2011). QBO and the mirror. What if..., <http://thecorpora.com/blog/?p=844>