

Bakalárska práca



České
vysoké
učení technické
v Praze

F3

Fakulta elektrotechnická
Katedra kybernetiky

Reprezentácia proprioceptívnych vstupov humanoidného robota iCub pomocou samoorganizujúcich sa máp

Nada Bednárová

Program: Otvorená Informatika
Obor: Informatika a počítačové vedy

Máj 2015

Vedúci práce: Mgr. Matěj Hoffmann, PhD.
Garant: Mgr. Michal Vavrečka, PhD.

PodĀakovanie / Prehlásenie

Chcem poĀakovať vĀetkým humanoidným robotom za to, že sú.

Prehlasujem, že som predloženú prácu vypracovala samostatne, a že som uviedla vĀetky použité zdroje v súlade s Metodickým pokynom o dodržiavání etických princípov pri príprave vysokoškolských záverečných prác.

.....
NaĀa Bednárová

V Prahe, 22. mája, 2015

Abstrakt / Abstract

Cielom tejto bakalárskej práce je vytvorenie biologicky inšpirovanej reprezentácie proprioceptívnych vstupov pre humanoidného robota iCub, študovaním a modelovaním procesov, ktoré sú zodpovedné za vývoj takýchto oblastí v primárnom somatosenzorickom kortexe mozgu u primátov. Nazbierané proprioceptívne dáta sú spracované pomocou rôznych typov populačného kódovania a následne pomocou Kohonovej architektúry Samoorganizujúcich sa máp. Výsledky sú porovnané medzi sebou a tiež s proprioceptívnymi reprezentáciami u primátov. Dôraz je kladený najmä na zachovanie topológie vstupných dát, na pomerné zastúpenie vstupných dát vo výslednej mape a na celkovú reprezentáciu proprioceptívnych vstupov naprieč všetkými stupňami voľnosti.

Kľúčové slová: Propriocepcia; kĺby; svalové vretienka; humanoidný robot; iCub; populačné kódovanie; Samoorganizujúce sa mapy; SOM

The goal of this thesis is to create a biologically inspired representation of proprioceptive inputs for the humanoid robot iCub, studying and modeling processes that are responsible for the development of such areas in primary somatosensory cortex of primate brain. The proprioceptive data collected is preprocessed using different kinds of population coding and then fed into a Self-Organizing Map to create the representation. Results for different parameter settings are compared with proprioceptive representation in primate brain, focusing on topological organization of the map, proportional representation of different values of inputs, and on the whole representation of proprioceptive inputs across all degrees of freedom.

Keywords: Proprioception; joints; muscle spindles; humanoid robot; iCub; population coding; Self-Organizing Maps; SOM

Title translation: Representation of proprioceptive inputs of the humanoid robot iCub using self-organizing maps

/ Obsah

1 Úvod a ciele práce	1
1.1 Prístup práce	1
1.2 Odkazy a zdroje	1
1.2.1 Stručný prehľad kapitol ...	2
2 Propriocepcia	3
2.1 Čo je propriocepcia	3
2.2 Proprioceptívne receptory	4
2.3 Mozog a propriocepcia	5
2.3.1 Typy neurónov	7
3 Reprezentácia propriocepce roboty iCub pomocou SOM	8
3.1 iCub a jeho pohybový aparát ...	8
3.2 Algoritmus učenia SOM	9
3.2.1 Priebeh učenia SOM	9
3.3 Problémy biologicky inšpiro- vanej reprezentácie a možné riešenia	11
3.3.1 Obmedzenia biologicky inšpirovanej reprezen- tácie u robota	11
3.3.2 Možné spôsoby repre- zentácie	12
4 Implementácia a testy	14
4.1 Implementácia	14
4.1.1 SOM toolbox	14
4.1.2 Vstupné dáta	15
4.2 Predspracovanie dát	15
4.2.1 Sigmoidy	16
4.2.2 Gaussove krivky	18
4.2.3 Lineárne funkcie	19
4.3 Problémy populačného kó- dovania	19
4.4 Algoritmus	22
4.5 Testovanie	22
5 Záver	26
5.1 Diskusia	26
Literatúra	28
A Zadanie	32
B Konštrukcia robota iCub	33
C Obsah CD	35

Tabuľky / Obrázky

3.1. Zoznam jednotlivých SV robota iCub.....9	2.1. Štruktúra svalového vretienka ...5
B.1. Kompletný zoznam SV robota iCub 34	2.2. Brodmannova oblasť 3a6
	3.1. Pohybová štruktúra robota iCub8
	3.2. Samoorganizujúce sa mapy 10
	4.1. Populačné kódovanie sigmoidami..... 17
	4.2. Populačné kódovanie Gaussovými krivkami..... 18
	4.3. Inicializácia SOM 20
	4.4. Priebeh učenia SOM 21
	4.5. Kernel Density Estimation 21
	4.6. Reprezentácia vstupných dát .. 23
	4.7. Topografické usporiadanie mapy 23
	4.8. Pomerné zastúpenie dát v mape 24
	4.9. Ukážka výslednej mapy 25
	B.1. Štupne voľnosti robota iCub... 33

Kapitola 1

Úvod a ciele práce

S pribúdajúcimi technológiami a s tým súvisiacim prehľbujúcim sa výskumom fungovania mozgu a neurovedy sa postupne rozširujú aj poznatky v oblasti spôsobu ľudského vnímania, reprezentácie vnemov, znalostí a spôsobu učenia. Pomocou týchto technológií sme zároveň schopní získané vedomosti prakticky uplatňovať a tým rozširovať ich možnosti napríklad v zmysle strojového učenia, rozpoznávania, optimalizácie a robotiky, pričom mnohé z používaných algoritmov sú inšpirované biológiou [1]. Rovnako aj cieľom tejto práce je vytvoriť reprezentáciu proprioceptívnych vstupov humanoidného robota iCub pomocou samoorganizujúcich sa máp, ktorá bude inšpirovaná biologickou reprezentáciou proprioceptívnych vstupov človeka v somatosenzorickom kortexe, čo by malo umožniť robotovi rozpoznávať polohu vlastných končatín a trupu podobne ako je to u človeka.

1.1 Prístup práce

Táto práca nadväzuje na bakalársku prácu a výskum Zdeňka Straku, ktorý pomocou upraveného algoritmu samoorganizujúcich sa máp (mSOM) vytvoril reprezentáciu taktilných vstupov humanoidného robota iCub [2]. V nasledujúcich kapitolách budem pomocou samoorganizujúcich sa máp vytvárať biologicky inšpirovanú reprezentáciu proprioceptívnych vstupov tohoto robota. Keďže o proprioceptii a jej reprezentácii u človeka doposiaľ nemáme také presné informácie ako o taktilných vstupoch a ich spracovávaní, pochopenie tejto problematiky bude vyžadovať o niečo rozsiahlejší výskum v oblasti neurológie. V úvode sa preto sústredím na čo najpresnejšie zistenie pôvodu proprioceptívnych vnemov u človeka a spôsobu ich prenosu, spracovania a reprezentácie v mozgu, čo by malo viesť k lepšiemu pochopeniu možností ako pristupovať k práci s týmto typom vnemov u robota. Zároveň na rozdiel od taktilných vstupov v [2] bude nutné viac pracovať s formátom dát, ktorý treba nejakým spôsobom zjednotiť. V literatúre sa najčastejšie ku kódovaniu komplexnejších parametrov pristupuje takzvaným populačným kódovaním, ktoré sme sa rozhodli otestovať aj v rámci reprezentácie proprioceptívnych vstupov.

1.2 Odkazy a zdroje

Informácie v tejto bakalárskej práci sú čerpané z rôznych odborných článkov a výskumov o proprioceptii, najmä v oblasti neurovedy [3–5] a populačného kódovania [6–8]. V rámci implementácie vychádzam z aktuálnej dokumentácie k robotovi iCub ¹⁾ a v Matlabe využívam doplnok SOM toolbox ²⁾ s dokumentáciou [9]. Dáta, na ktorých testujem navrhovanú reprezentáciu, zozbieral vedúci práce Matěj Hoffmann.

¹⁾ http://wiki.icub.org/wiki/Main_Page

²⁾ <http://www.cis.hut.fi/somtoolbox/>

■ 1.2.1 Stručný prehľad kapitol

- V **kapitole 1** sú popísané zdroje, z ktorých v práci čerpám a na ktoré práca nadväzuje, vymedzenie cieľov práce, jej prístup a tiež stručný obsah a prehľad kapitol.
- **Kapitola 2** vysvetľuje základný pojem *propriocepcia* a jej reprezentáciu a význam u človeka.
- V **kapitole 3** popisujem štruktúru kĺbov robota iCub, algoritmus učenia pomocou samoorganizujúcich sa máp SOM, konkrétnu problematiku práce a tiež spôsoby jej možného riešenia.
- **Kapitola 4** rozoberá konkrétnu implementáciu a testovanie navrhnutých riešení.
- A v **závere 5** sú zhrnuté výsledky, problémy, prínos práce a možné spôsoby jej ďalšieho rozšírenia, prípadne využitia.

V **prílohe B** uvádzam výpis všetkých stupňov voľnosti robota iCub s ich rozsahmi a v **prílohe C** sa nachádza obsah priloženého CD.

Keďže prevažná väčšina odbornej literatúry k týmto témam je v anglickom jazyku, za dôležitými pojmami uvádzam vždy v zátvorke ich anglický ekvivalent. Zároveň pre lepšiu predstavu biologických a neurologických súvislostí prikladám k dôležitým častiam ilustratívne obrázky.

Kapitola 2

Propriocepcia

V tejto kapitole popíšem význam slova propriocepcia, čo všetko tento pojem u človeka zahŕňa, akým spôsobom sú proprioceptívne informácie zachytávané na úrovni receptorov a tiež ďalej vyhodnocované v somatosenzorickom kortexe šedej mozgovej kôry.

2.1 Čo je propriocepcia

Výraz propriocepcia prvýkrát použil C. S. Sherrington vo svojom článku [10] v roku 1907, kde rozdelil senzorické receptory do troch kategórií :

- **Exterocepcia**, ako schopnosť tela reagovať na okolité podnety, zahŕňa receptory reagujúce na mechanický kontakt, chlad, teplo, svetlo, zvuk a iné mechanické a chemické receptory, ktoré zbierajú informácie z vonkajšieho prostredia
- **Interocepcia**, ako schopnosť zbierať informácie z vnútorného prostredia človeka (napríklad orgánov), zahŕňa receptory reagujúce na chemické vzruchy, ktoré podnecujú napríklad chuť, zažívanie či bolesť
- **Propriocepcia**, reakcia tela na vlastné podnety (vnímanie vlastného pohybu, postavenia tela, vynaloženej sily, rýchlosti, smeru)

Propriocepcia teda vo všeobecnosti zahŕňa schopnosť človeka vnímať a koordinovať statické postavenie vlastných končatín a trupu (limb-position sense), ich pohyb (kinesthesia), rýchlosť a smer, bez využitia zraku [3]. Vďaka proprioceptii sme schopní vnímať relatívnu polohu nášho tela, vzájomnú polohu jeho susedných častí, silu, ktorou pôsobia, ale aj úsilie, ktoré musíme na ich pohyb vynaložiť. Okrem iného je propriocepcia spolu s vestibulárnym aparátom zodpovedná za schopnosť udržať stabilitu, rovnováhu a vzpriamené držanie tela.

V užšom slova zmysle je propriocepcia chápaná ako schopnosť tela vyhodnocovať proprioceptívne podnety a reagovať na ne na bunkovej úrovni, teda zbierať informácie o pozícii tela zo svalových vlákien, šliach, kĺbov a taktilných receptorov a prenášať ich do somatosenzorického kortexu šedej mozgovej kôry, kde sú tieto signály ďalej reprezentované a vyhodnocované [5]. Rozsah pohybu a tým aj rozsah takto definovaných proprioceptívnych dát je pritom u človeka daný pohybovým rozsahom jednotlivých kĺbov.

Z hľadiska pohybu ako takého sú najdôležitejšie synoviálne kĺby, ktoré sú vďaka synoviálnej vrstve ¹⁾ voľne pohyblivé. Synoviálne kĺby sa navzájom líšia rôznymi tvarmi a z toho vyplývajúcimi osami pohybu, podľa čoho ich delíme na niekoľko základných typov, pre predstavu uvádzam niekoľko typických príkladov:

- Valcový kĺb (Hinge joint) - napr. v lakti - umožňuje pohyb v jednej osi

¹⁾ Synoviálna vrstva je riedka väzivová membrána, ktorá pokrýva zakončenia kostí/chrupavky a vyplnía celé vnútro kĺbu. Zároveň produkuje kĺby maz, takzvanú synoviálnu tekutinu, vyživujúcu chrupavku a znižujúcu trenie, vďaka čomu sú synoviálne kĺby voľne pohyblivé.

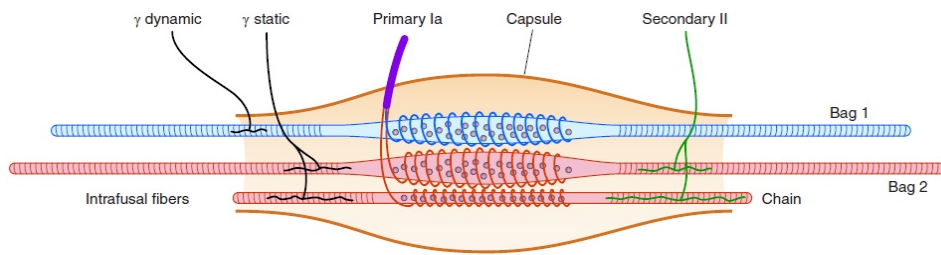
- Sedlový kĺb (Saddle joint) - napr. palec na ruke - pohyb v dvoch osiach
- Gulovitý kĺb (Ball-and-socket joint) - napríklad ramenný kĺb - umožňuje pohyb a rotáciu vo viac ako dvoch osiach pohybu

Z hľadiska propriocepcie je dôležité si tieto rozdiely v počte osí a rozsahoch pohybu jednotlivých kĺbov uvedomiť.

2.2 Proprioceptívne receptory

Pohyb je u človeka spôsobovaný kontrakciou svalov, ktoré svojím skracovaním postupne menia pozíciu tela a teda uhly jednotlivých kĺbov [3]. Dĺžku svalu a uhol, o ktorý sa príslušný kĺb otočí, pritom spája nelineárny vzťah, ktorý sa dá vyjadriť približne kosínovou vetou [11]. Túto súvislosť však pre zjednodušenie v rámci našej reprezentácie zanedbávame a sústredíme sa priamo na vzťah zmeny uhlu a frekvencie pálenia neurónov, ktoré pozíciu daného kĺbu kódujú. Receptory, ktoré sú u človeka zodpovedné za propriocepciu, nazývané proprioceptory, sa teda nachádzajú v svaloch, ale aj šlachách a tiež kĺboch a pokožke, ktorých deformácia je týmto pohybom spôsobená, a zbierajú informácie o svalovom napätí (tonus), deformácii kĺbových púzdiar, napínaní a uvoľnení pokožky, o rýchlosti pohybu, či vynaloženej sile [5]. Za proprioceptory sú považované hlavne :

- **Svalové vretienka** (muscle spindles) - Svalové vretienka, vid' obrázok 2.1, sú sensorické receptory zbierajúce informácie o dĺžke svalu a rýchlosti jeho pohybu. Nachádzajú sa v svalu paralelne s extrafuzálnymi svalovými vláknami (vykonávajúcimi samotnú kontrakciu) a sú zložené z intrafuzálnych vlákien dynamických, statických a reťazových (dynamic bag1, static bag2, chain) uzavrených vo väzivovom obale a niekoľkých typov aferentných a eferentných sensorických vlákien [3, 5] Propriocepcia nie je vymedzená len na pohyb, proprioceptívne informácie o tele vyhodnocujeme neustále, aj v statickej pozícii tela. Najväčší proprioceptívny význam preto majú sensorické vlákna napojené na statické intrafuzálne vlákna a to konkrétne primárne svalové vretienka typu Ia a sekundárne typu II [5]. Oba typy nesú informáciu o dĺžke statického svalu, hoci typ Ia, ktorý zároveň leží na dynamických intrafuzálnych vláknach, reaguje tiež na pohyb svalu, a teda nesie informácie aj o rýchlosti zmeny jeho dĺžky. Nervové vlákna, ktoré signály zo svalových vretienok prenášajú ďalej do centrálného nervového systému (CNS), menia frekvenciu pálenia v závislosti na dĺžke svalu. Môžeme teda povedať, že svalové vretienka a ich aferentné neuróny kódujú dĺžku svalov a tým zároveň uhly príslušných kĺbov, prípadne rotáciu kĺbu v určitom smere, čo je jedným z dôvodov pre všeobecný príklon k názoru, že svalové vretienka sú hlavnými proprioceptormi [5, 8]. Podľa [12–14] je toto kódovanie prevažne jednosmerné, čo znamená, že nervové vlákna pália iba v určitom smere rotácie a to pri natahovaní svalu, v ktorom sa nachádzajú [14]. Rotácia jedného kĺbu je však zakódovaná veľkým počtom neurónov (takzvanými populáciami neurónov), preto je na mieste otázka, či všetky aferentné vlákna svalových vretienok kódujú pohyb a rozsah kĺbu rovnako. Existuje uznávaný predpoklad [7], že sensorické vlákna v každom svalu majú svoj 'preferovaný rozsah'/'smer' pohybu/rotácie, alebo preferovanú pozíciu kĺbu (muscle's preferred direction), v ktorej aferentné neuróny sensorických vlákien svalu pália najviac. Podľa [7] však ani sensorické vlákna v rámci jedného svalu nereagujú rovnako na rovnaké rozsahy, či smer pohybu, ale ich 'preferované rozsahy' sa navzájom prekrývajú a po sčítaní dajú výsledný 'smer' celého svalu, teda rozsah, pre ktorý populácia aferentných neurónov daného svalu páli najviac. Pre kĺby s viacerými osami rotácie sú tieto smery 'viacdimenzionálne' [7, 14].



Obrázok 2.1. Štruktúra svalového vretienka [5]

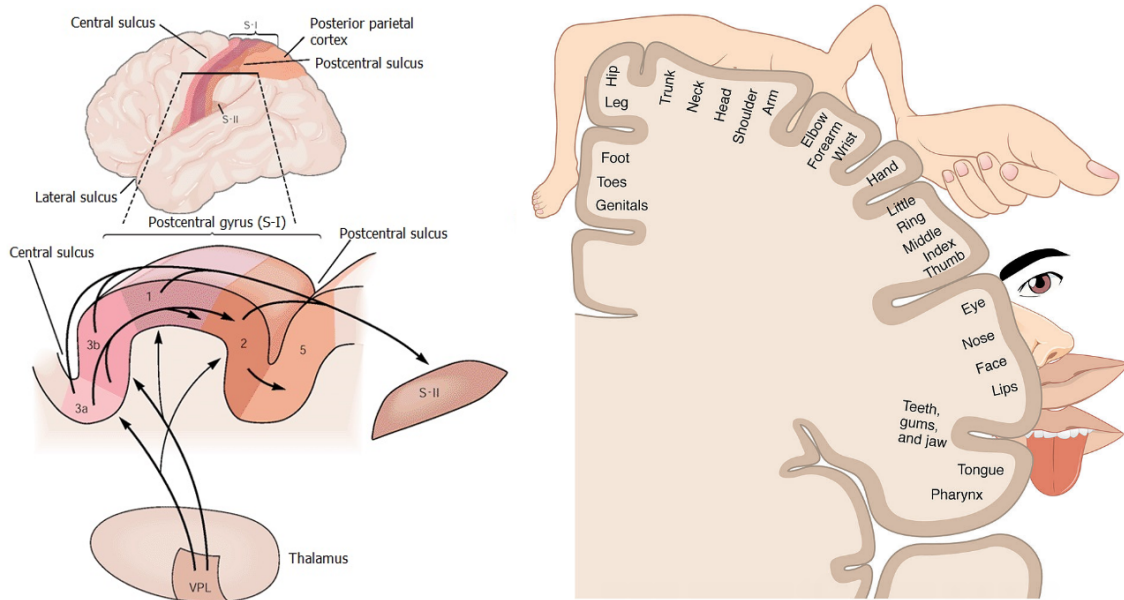
- Golgiho šlachové telieska** (Golgi tendon organs) - Tieto proprioceptory inervované aferentnými neurónmi Ib sa nachádzajú v šľachách tesne pri spoji so svalom. Keďže sú k svalovým vláknam pripojené 'sériovo', reagujú najmä na napätie šľachy a silu, ktorou je sval sťahovaný [3]. Keďže prevažná väčšina štúdií o proprioceptcii sa sústreďuje na svalové vretienka ako na primárne proprioceptory, štúdie o prispievaní Golgiho šlachových teliesok k proprioceptívnemu vnímaniu nie sú jednotné. Podľa [8] nereagujú na pozíciu končatín, ani na pasívny pohyb svaly. Naopak podľa ([3] s. 725) sú voľné nervové zakončenia aferentných vlákien Ib vpletené medzi kolagénové vlákna šlachového telieska, čo spôsobuje, že na skrátenie svaly, a teda natiahnutie šľachy, aferentné vlákna reagujú so zvýšenou citlivosťou.
- Nervové vlákna kĺbových púzdiar** (joint afferents) - Nervové vlákna inervujúce kĺbové púzdra sa aktivujú najmä v krajných polohách kĺbu [5]. Často ide o pohyb v extrémoch rozpätia kĺbu, prípadne o silný tlak vyvíjaný priamo na kĺbové púzdro, čo spôsobuje reakciu na pohyb vo viacerých osiach rotácie a tiež vo viacerých smeroch [8].
- Taktilné receptory** (cutaneous receptors) - Ruffiniho telieska (Ruffini endings) a Valter-Paciniho telieska (Lamellar/Pacinian corpuscles) sú citlivé na vibrácie a tlak na pokožke, takže sú schopné detekovať dotyk a deformáciu pokožky pri pohybe. V proprioceptcii majú len okrajovú úlohu.

Proprioceptívne signály z týchto receptorov prechádzajú ďalej rôznymi cestami miechy do mozgu, kde sú priamo reprezentované, alebo ďalej transformované, spájané s informáciami z iných receptorov a vyhodnocované.

2.3 Mozog a propriocepcia

Za hlavnú reprezentáciu proprioceptcie v mozgu primátov je považovaná takzvaná Brodmannova oblasť (BA) 3a, ktorá sa nachádza v temennom laloku (parietal lobe) v časti somatosenzorický kortex (somatosensory cortex), najčastejšie na postcentrálnej závitke (postcentral gyrus) mozgovej kôry (cerebral cortex) tesne za centrálnou brázdou (central sulcus). Umiestnenie, rovnako ako aj usporiadanie oblasti 3a, sa však u jedincov môže mierne odlišovať [4]. Hlavnou cestou, ktorou do tejto oblasti proprioceptívne informácie prichádzajú, sú neuróny miechových uzlín - spinálnych ganglií (dorsal root ganglion neurons). Spolu s ostatnými somatosenzorickými informáciami prechádzajú cez časť miechy zvanej nucleus cuneatus do kontralaterálnej časti medial lemniscusu, čo znamená, že už v mieche sa tieto informácie z jednotlivých častí tela prekrížia a prejdú do opačnej, teda z ľavej do pravej (a naopak) časti CNS. Do samotnej oblasti 3a prichádza takýto aferentný signál z miechy takzvaným medzimotozgovým lôžkom (thalamus). V niektorých výskumoch [15] sú uvedené zložité mechanizmy prepojenia jednotlivých častí somatosenzorického kortexu a sú skúmané takzvané multimodálne súvislosti [16]

jednotlivých oblastí, čo v našom prípade znamená, že do Brodmannovej oblasti okrem proprioceptívnych vstupov zo spinálnych ganglií a thalamu prichádzajú ešte informácie rôzneho charakteru z ďalších oblastí mozgu, napríklad z BA 3b, alebo mozočku (cerebellum), takzvaným dorsálnym spinocerebellárnym traktom. Pre zjednodušenie však v rámci tejto práce uvedené súvislosti zanedbáme a budeme sa sústrediť čisto na proprioceptívne informácie prechádzajúce do BA 3a spinálnymi gangliami.



Obrázok 2.2. Somatosenzorický kortex S1–vľavo hore je bledou ružovou farbou označaná približná poloha Brodmannovej oblasti 3a a vľavo dole zväčšený postcentrálny závit s oblasťami 3a, 3b, 1 a 2 a šípkami, ktoré označujú ako sú tieto oblasti prepojené a kam informácie z nich putujú ďalej. Na pravej strane je zobrazený zjednodušený model reprezentácie jednotlivých častí tela v S1 ([3] s. 385, [17]).

Somatosenzorický kortex S1, teda oblasť mozgu vyhodnocujúca fyzické zmyslové vnemy, zbiera informácie zo somatosenzorických receptorov, teda receptorov zodpovedných za taktilné vnímanie, termocepciu (teplocitlivosť), nocicepciu (vnímanie bolesti) a propriocepciu. Podľa Brodmannovho členenia mozgovej kôry ([3] s. 13) sa táto oblasť delí na Brodmannove oblasti 3, 1 a 2 (v poradí odpredu-dozaďu/anterior-posterior), pričom BA 3 je ďalej rozdelená na oblasť 3a a 3b somatosenzorického kortexu. Zatiaľ čo BA 3b spracováva a reprezentuje najmä taktilné vstupy, pre propriocepciu ako takú má najväčší význam Brodmannova oblasť 3a. Podľa [18] somatosenzorické dáta prichádzajú najskôr práve do týchto oblastí (3a a 3b), na základe čoho dostali názov primárny somatosenzorický kortex, a až odtiaľ postupne prechádzajú do BA 1 a 2 a sú ďalej spracovávané a vyhodnocované spolu s ďalšími vnemami, napríklad vestibulárnymi, viď obrázok 2.2.

Každá z oblastí primárneho somatosenzorického kortexu obsahuje reprezentáciu jednotlivých častí ľudského tela, podobnú tej na obrázku 2.2 vpravo, prevažne v topologickom usporiadaní. Reprezentácia taktilných vstupov v oblasti 3b má pomerne presné usporiadanie, pričom oproti reálnej topológii obsahuje dve významné diskontinuity – ruky sú reprezentované blízko tváre a nohy vedľa genitálií. Spolu s topologickým usporiadaním je pre oblasť 3b typické pomerne zastúpenie jednotlivých častí tela, ktoré odpovedá rôznej hustote ich inervácie, teda napríklad ruky majú v mape väčšie zastúpenie v pomere k iným častiam tela, napríklad trupu, v skutočnosti. Napriek podobnosti reprezentácií v oblasti 3a a 3b, oblasť 3a nemá tak striktné dané usporiadanie, teda u

každého jedinca sa môže mierne odlišovať a tiež nie vždy zachováva presné topologické usporiadanie. Tiež nie je typické pomerné zastúpenie na základe hustoty inervácie ako v 3b, ale skôr miery ich používania [4]. Výsledkom rovnakej štúdie je tiež pozorovanie, že receptívne polia neurónov v oblasti 3a sú ďaleko väčšie než v oblasti 3b.

■ 2.3.1 Typy neurónov

Podľa najnovšieho výskumu [16], ktorý vychádza z predpokladu, že jednotlivé reprezentácie a kódovania v mozgu majú multimodálnu povahu, skúma tiež rozdielnu povahu jednotlivých neurónov v rámci jednej oblasti/reprezentácie. Z experimentov vyplynulo, že neuróny reprezentujúce proprioceptívne vstupy ruky v mozgu, sa dajú rozdeliť na tri základné typy podľa veľkosti receptívnych polí (receptive fields), respektíve oblastí, ktoré reprezentujú a typu informácií, na ktoré v rámci danej oblasti reagujú:

- **Neuróny reagujúce na pozíciu jedného prsta** (single-digit position-scaled neurons), ktorých je zhruba 60% a reagujú na zmenu pozície/uhlu jedného kĺbu, prevažne lineárnym zvyšovaním, resp. znižovaním frekvencie pálenia.
- **Neuróny reagujúce na koaktiváciu viacerých prstov** (multi-digit position-scaled neurons), napríklad na vzdalovanie prstov navzájom, alebo na vzdalovanie a súčasné pokrčenie iného prsta. Odpoveď je naprieč pozíciami prevažne tiež lineárna.
- **Neuróny, ktoré kódujú postavenie väčšej časti tela** (posture-selective neurons) sú charakteristické nelineárnymi súvislosťami medzi reakciami na pohyb, resp. rotáciu príslušných kĺbov.

V niektorých článkoch sa môžeme tiež stretnúť s neurónmi reagujúcimi na tzv. ‘end-point position’, čo je pozícia konca končatiny, najčastejšie dlane, alebo chodidla, v priestore vzhľadom k inej, referenčnej časti tela, napríklad trupu. Tieto informácie však podľa výskumu [19] prechádzajú už spomínaným spinocerebellárnym traktom a teda nie je zrejmé či sú súčasťou proprioceptívnej reprezentácie v Brodmannovej oblasti 3a, preto sa ich modelovaniu v tejto práci nebudeme venovať.

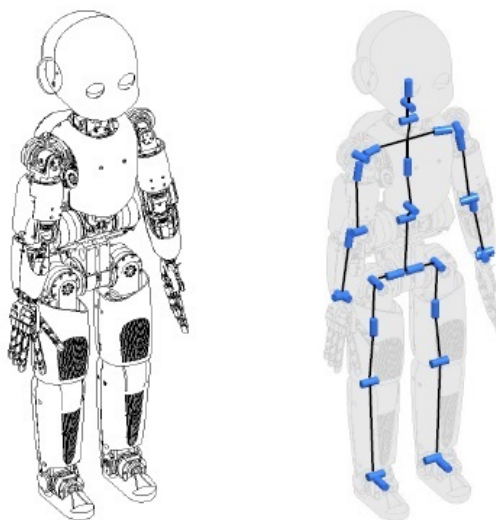
Kapitola 3

Reprezentácia propriocepce robota iCub pomocou SOM

V tejto kapitole popíšem pohybovú štruktúru robota iCub, algoritmus učenia samooorganizujúcich sa máp a následne rozoberiem problematiku biologicky inšpirovanej reprezentácie proprioceptívnych vstupov robota iCub pomocou týchto máp spolu s možnými prístupmi k danej problematike a navrhovanými spôsobmi jej riešenia.

3.1 iCub a jeho pohybový aparát

iCub je približne meter vysoký a 24kg vážiaci open-source humanoidný robot zostrojený v Italian Institute of Technology v spolupráci s niekoľkými európskymi univerzitami a laboratóriami za účelom výskumu umelej inteligencie a ľudskej kognície. ¹⁾ Vďaka jednoduchej dostupnosti, prehľadnej dokumentácii, množstvu prispievateľov a tiež vďaka možnosti využiť rovnako voľne dostupný iCub simulátor, sa robot iCub stal obľúbeným prostriedkom výskumu v rôznych oblastiach kybernetiky, robotiky a umelej inteligencie na univerzitách, či v rôznych výskumných centrách.



Obrázok 3.1. Pohybová štruktúra robota iCub [20].

Ako je popísané v kapitole 2, jednotlivé kĺby sa u človeka odlišujú množstvom osí rotácie a ich rozsahom. Podobne je to u robota iCub. Aby sa jeho pohybové možnosti čo najviac podobali ľudským, každý kĺb môže rotovať v jednej a viac osiach, pričom rozpätie jednotlivých osí sa líši. Jedna os rotácie teda zodpovedá jednému stupňu voľnosti (degree of freedom – DOF). Pre sprehľadnenie budem preto ďalej používať výraz *kĺb* v biologickom zmysle a výraz *stupeň voľnosti*, alebo *SV* pre jednotlivé osi rotácie.

¹⁾ http://wiki.icub.org/wiki/Main_Page

Pohybový aparát robota iCub je zložený z motorov napojených na jednotlivé stupne voľnosti, ktorých je celkovo 53 (6x hlava, 3x trup, 16+16x ruky a 6+6x nohy). Pre predstavu uvádzam príklad v tabuľke 3.1, podrobná tabuľka so všetkými SV a ich rozsahmi sa nachádza v prílohe B.

časť tela	kĺb	stupeň voľnosti	od	do	rozpätie
trup	pás	otočenie do strany	-50	50	100
		úklon do strany	-30	30	60
		záklon/predklon	-10	70	80
ruka	rameno	za-/predpaženie	-95	10	105
		upaženie	0	160,8	160,8
		rotácia paže	-37	80	117
	lakeť	pokrčenie/vystretie	15,5	106	121,5

Tabuľka 3.1. Príklad jednotlivých stupňov voľnosti kĺbov robota iCub v stupňoch.

Ako môžeme v uvedenej tabuľke 3.1 vidieť, stupne voľnosti kĺbov robota iCub majú rôzne osi a rozsahy rotácie, pričom minimálny rozsah je 44 stupňov u pohybu členka do strán a maximálny 270 stupňov pre pokrčenie prstenníka s malíčkom. Jednotlivé rozsahy nie sú symetrické okolo nuly. Podobne ako u človeka, každý stupeň voľnosti má zároveň svoju ‘normálnu polohu’, ktorá určuje základné/východzie postavenie tela (nie je vždy v nule). Rozsahy a východzie polohy kĺbov nie sú presne dané hodnotami týchto parametrov u človeka, ale sú prispôsobené technickým možnostiam robota.

3.2 Algoritmus učenia SOM

Samoorganizujúce sa mapy (Self-Organizing Maps), podľa svojho autora tiež nazývané Kohonenove mapy, sú druhom umelých neurónových sietí, ktorých učenie prebieha ‘bez učiteľa’. Na rozdiel od ostatných podobných algoritmov sú však SOM charakteristické zachovávaním topologickej štruktúry vstupných dát, čo znamená, že podobné vstupné dáta sú po naučení vo výslednej mape reprezentované susediacimi neurónmi, teda že vstupné dáta a výstupné neuróny majú podobné topologické vzťahy. [21] Algoritmus SOM je tiež schopný redukovat’ spojité, alebo diskrétny vstupný priestor vysokej dimenzie na diskrétny výstupný priestor nižšej dimezie [2, 22].

Jednotlivé neuróny výslednej naučenej mapy reprezentujú špecifické súvislosti, ‘vzory’, ‘triedy’, alebo ‘prototypy’ vstupných dát. Samoorganizujúce sa mapy sú preto využívané najmä v kontrole procesov, telekomunikáciách, robotike, alebo rozpoznávaní, kde dosahujú dobré výsledky [22].

3.2.1 Priebeh učenia SOM

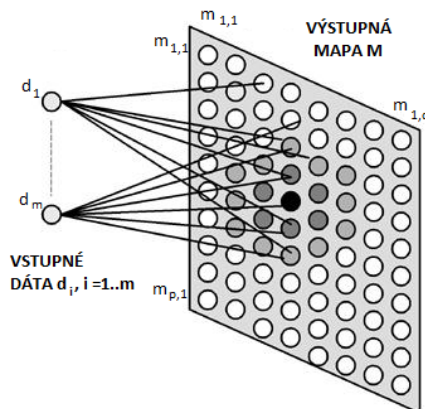
Samotný algoritmus, tak ako bol navrhnutý a popísaný Kohonenom v [21], je do veľkej miery všeobecný a má viacero modifikácií. Pre ilustráciu a prehľadnosť budem popisovať všeobecný priebeh učenia samoorganizujúcich sa máp, ktorý je používaný najčastejšie a vyhovuje tiež našim podmienkam, hoci typ a formát vstupných dát, ako aj spôsob počítania vzdialeností jednotlivých neurónov a iné parametre učenia, sa môžu v rôznych modifikáciách odlišovať.

Vstupom algoritmu SOM je v našom prípade matica $D_{i,j}$ veľkosti $m \times n$, ktorej riadky d_i odpovedajú jednotlivým tréningovým dátam, pričom n je počet prvkov vstupného

vektora a m je počet týchto vektorov.

$$D_{i,j} = \begin{pmatrix} d_{1,1} & d_{1,2} & \dots & d_{1,n} \\ \vdots & & & \vdots \\ d_{m,1} & d_{m,2} & \dots & d_{m,n} \end{pmatrix}, d_{i,j} \in [0, 1]$$

Vstupné dáta sú bezprostredne spojené s výstupnou vrstvou (viď obrázok 3.2), pričom každý neurón $m_{p,q}$ výslednej mapy $M \in R^{p \times q \times j}$ je spojený s každým prvkom j vstupného vektora d_i a $(p \times q)$ je veľkosť výslednej mapy. Neuróny v mape sú tak reprezentované vektormi váh, ktoré priamo predstavujú určitý charakteristický ‘vzor’, alebo ‘prototyp’ vstupných dát.



Obrázok 3.2. Samoorganizujúce sa mapy – obrázok je pre zachovanie prehľadnosti zjednodušený, v skutočnosti je každý vektor na vstupe je prepojený s každým neurónom výslednej mapy M.

Na začiatku učenia sa váhy neurónov výslednej mapy inicializujú, najčastejšie náhodným generovaním, v rovnakom rozsahu ako možný rozsah jednotlivých prvkov vstupných dát, v tomto prípade $m_{p,q,j} \in [0, 1]$. Následne prebieha samotná fáza učenia a to v niekoľkých krokoch:

1. Najskôr je z matice vstupných dát náhodne, alebo v poradí vybraný jeden riadok d_i , odpovedajúci vstupnému vektoru. Pre zvolený vektor je následne vypočítaná vzdialenosť od každého neurónu výstupnej mapy v zmysle napríklad euklidovskej normy, alebo skalárneho súčinu, ktorá určí ako veľmi sa jednotlivé neuróny ‘podobajú’ na vstupný vektor.
2. Na základe vzdialeností jednotlivých neurónov je potom určený víťazný neurón (best matching unit – BMU) $m_{p,q}$, teda taký, ktorý sa na vstupný vektor ‘najviac podobá’, respektíve ktorý má najmenšiu euklidovskú vzdialenosť od vstupného vektora,

$$c = (p^*, q^*) = \operatorname{argmin}_{(p,q)} (\|m_{p,q} - d_i\|_2^2)$$

alebo najväčší skalárny súčin

$$c = (p^*, q^*) = \operatorname{argmax}_{(p,q)} (m_{p,q} \cdot d_i)$$

3. V nasledujúcom kroku prebieha takzvaná fáza adaptácie víťazného neurónu a jeho okolia, kedy sa okolie víťazného neurónu ‘priblíži’ tomuto neurónu v zmysle funkcie vzdialenosti, tak ako bola vydefinovaná vyššie, a to podľa funkcie okolia:

$$h_{cs}(t) = e^{-r_{cs}^2 / 2\sigma_t^2}$$

$$m_s(t+1) = m_s(t) + \alpha(t)h_{cs}(t)[d_i(t) - m_s(t)]$$

Pričom $r_{cs} = \|c-s\|$ je vzdialenosť pozície $c = (p^*, q^*)$ víťazného neurónu od ostatných prvkov mapy $s = (p, q)$ [9].

4. Miera ‘približovania’ sa, respektíve rýchlosť učenia (learning rate) $\alpha(t)$ sa pritom znižuje s časom napríklad podľa lineárnej funkcie:

$$\alpha(t) = \alpha_0 \cdot (1 - t/T)$$

kde T je dĺžka učenia a α_0 počiatočná rýchlosť učenia. Pre bližšie informácie viď [9].

3.3 Problémy biologicky inšpirovanej reprezentácie a možné riešenia

V tejto časti popíšem konkrétne problémy, na ktoré pri snahe o vytvorenie biologicky inšpirovanej reprezentácie propriocepcie môžeme naraziť a tiež možné riešenia vychádzajúce z výskumu propriocepce a jej reprezentácie u človeka. Zároveň popíšem a zdôvodním reprezentáciu, pre ktorú sme sa v tejto práci rozhodli.

3.3.1 Obmedzenia biologicky inšpirovanej reprezentácie u robota

Cieľom tejto práce, ako bolo spomenuté v úvode, je vytvoriť reprezentáciu proprioceptívnych vstupov pre robota iCub inšpirovanú reprezentáciou propriocepce u človeka. Z toho vyplýva niekoľko problémov :

- Pohyb človeka je komplexná problematika, ktorá zahŕňa zložitú mechaniku a množstvo rozličných elektrických, či chemických reakcií. Ako uvádzam v kapitole 2, proprioceptívne informácie pochádzajú z niekoľkých typov receptorov (svalové vretienka, Golgiho šlachové telieska, kĺbové púzdra a pod.), ktoré zbierajú o pozíciách tela dáta rôzneho typu. U robota iCub je mechanika pohybu založená na rotáciách jednotlivých stupňov voľnosti. Množstvo rôznych proprioceptívnych vstupov robota je teda u robota iCub obmedzené na informácie o veľkosti uhlov jeho stupňov voľnosti. Aby sme reprezentáciu týchto vstupov čo najviac priblížili ľudskej, budeme musieť potrebné parametre simulovať na základe informácií, ktoré máme.
- Podľa ([23] s. 455) je počet stupňov voľnosti u človeka 244 a počet prične pruhovalných svalov (skeletal muscles), v ktorých sa nachádzajú svalové vretienka, je 630. U robota iCub máme len stupne voľnosti, ktorých je spolu 53 (viď príloha B). Nie je teda možné, aby výsledná mapa reprezentujúca jednotlivé časti tela bola tak presná a detailná ako u človeka.
- Propriocepčia, na rozdiel od napríklad taktilného vnímania, nie je tak detailne preskúmaná. Rôzne výskumy v tejto oblasti dospievajú často k protichodným názorom a je nutné zvažovať ich pravdivosť a relevantnosť na základe postupov, metód a techniky, ktoré pri výskume, alebo argumentácii používajú. Presný spôsob vytvárania proprioceptívnej reprezentácie v mozgu zatiaľ nie je dostatočne detailne zmapovaný a popísaný.
- Zároveň podľa výskumu [4] nie je reprezentácia v Brodmannovej oblasti 3a zďaleka tak striktná a presne daná ako reprezentácia taktilných vstupov v oblasti 3b. Napriek podobnosti napríklad v topologickom usporiadaní a nespojitostiach, sa usporiadanie a zastúpenie jednotlivých častí tela v oblasti 3a u rôznych jedincov môže mierne odlišovať, napríklad podľa miery využívania danej časti tela.

3.3.2 Možné spôsoby reprezentácie

- V prvom rade je nutné zamyslieť sa nad samotným formátom dát. Dáta, ktoré zo senzorov robota iCub prichádzajú, označujú uhly v stupňoch a nesú informáciu o rotácii pre každý stupeň voľnosti zvlášť. Keďže rozsahy jednotlivých SV sú rôzne (v rozmedzí od -125 do 270 a rozsahu od 44 do 270 stupňov) a keďže algoritmus SOM počíta vzdialenosti vstupných dát od jednotlivých neurónov pomocou euklidovskej vzdialenosti, bude nutné rozsahy zjednotiť a teda nejakým spôsobom normovať, v opačnom prípade by boli pri hľadaní najmenej vzdialenosti zvýhodnené stupne voľnosti s menším rozsahom. Jednoduché normovanie v zmysle lineárneho namapovania pôvodného rozsahu v stupňoch na interval $[0, 1]$ však nebude pre naše účely postačujúce. Ak by sme dáta normovali týmto spôsobom, mali by sme pre každý SV jeden pomyselný neurón, ktorý by uhol kódoval priamo priemernou frekvenciou pálenia, teda čím väčší by bol uhol, tým vyššia by bola frekvencia pálenia. Znamenalo by to, že vo vychýlenej pozícii by bola hodnota znormovaná na nulu a vstupný neurón nepálil, čo v biológii nie je bežné, pretože proprioceptívne informácie o pozícii a postavení tela, respektíve jeho častí, z receptorov dostávame nepretržite.
- V biológii tiež nie je bežné, aby boli napríklad všetky proprioceptívne informácie z jedného valcového kĺbu s jedným stupňom voľnosti kódované výhradne jedným neurónom samotnou frekvenciou jeho pálenia. Ako bolo spomenuté v kapitole 2, u človeka je informácia o rotácii kĺbu kódovaná väčšinou viacerými neurónmi rôzneho typu. Rozličné rozsahy uhlov, prípadne receptívne polia, na ktoré jednotlivé neuróny reagujú, sa pritom prekrývajú, a tak spoločne kódujú celý rozsah pohybu danej časti tela. Tento prístup sa nazýva populačné kódovanie a je v biológii a tiež pre biologicky inšpirované algoritmy ako neurónové siete bežný, preto sme sa pre rovnaký prístup rozhodli aj v tejto práci.
- Neuróny v populáciách prijímajúcich informácie z proprioceptívnych senzoričných vlákien typicky nereagujú na celý rozsah, ktorý kódujú, rovnako. Na rôzne hodnoty vstupných dát každý neurón reaguje rôznou frekvenciou pálenia. Táto vlastnosť sa dá modelovať rôznymi funkciami, najčastejšie:

- **Lineárne funkcie** Podľa [6] ide popri Gaussových krivkách o jedno z najčastejšie používaných kódovaní pre modelovanie reakcií mozgových buniek. Pri rôznych parametroch a a ofsetoch b pre vstupné dáta x reprezentujúce v našom prípade jednotlivé pozície dostaneme zakódovanie jedného neurónu v populácii ako:

$$y = (a \cdot x) + b$$

- **Gaussove krivky** sú spolu s lineárnym kódovaním najtypickejšie pre populačné kódovanie pozícií [6]. Pre stred μ a parameter σ , reprezentujúci vzdialenosť stredu od inflexného bodu, dostaneme pre rovnaké pozície x zakódovanie:

$$y = e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Gaussovské populačné kódovanie je možné tiež použiť v binarizovanej verzii [24].

- **Sigmoidy** sú ďalším typom kódovania, ktorý je možné použiť pre modelovanie kódovania nervových signálov. Tým, že monotónne rastú, resp. klesajú, je možné nimi zároveň simulovať postupné zvyšovanie, resp. znižovanie frekvencie pálenia [25].

$$y = \frac{1}{1 + e^{\text{sgn} \cdot (-x + \text{offset})}}$$

Parameter *sgn* predstavuje orientáciu sigmoidy – negatívnu, alebo pozitívnu, *offset* reprezentuje pozíciu inflexného bodu a *x* sú vsupné hodnoty.

- V ľudskom tele väčšina svalov, ktorých kontrakcia spôsobuje rotáciu nejakého kĺbu, funguje v antagonistických pároch. To znamená, že jeden sval svojím skracovaním uhol znižuje a druhý zvyšuje. Rotáciu kĺbov s väčším počtom osí rotácie spôsobujú väčšinou viac ako len dva svaly a teda aferentné neuróny svalových vretienok v jednotlivých svaloch reagujú zároveň na rozsah v niekoľkých osiach rotácie, čím kódujú informáciu o viacrozmernom pohybe [7]. Keďže (ako bolo bližšie popísané v kapitole 2) svalové vretienka pália pri predlžovaní svalu, ich aferentné neuróny budú reagovať na jeden smer pohybu/rotácie kĺbu. Z toho vyplýva, že okrem samotného kódovania aktuálnej pozície kĺbu proprioceptory kódujú aj smer pohybu. Keďže u robota iCub je konštrukčne každá os rotácie separátnym stupňom volnosti podobným valcovému kĺbu u človeka, zjednodušíme náš model na jednotlivé separátne SV, čím dostávame možnosť kódovať každý SV dvomi antagonistickými svalmi, z ktorých každý nesie informáciu o jednom smere pohybu. Z hľadiska populačného kódovania dáva takýto prístup zmysel pri kódovaní sigmoidami, kde by pozitívne sigmoidy mohli predstavovať jeden sval pomyselného antagonistického páru a negatívne orientované druhý sval/smer pohybu.
- Podľa [26] sú všetky svaly, svalové vretienka a ich nervové pripojenie na miechu vyvinuté už v prenatalnom štádiu okolo 25. týždňa vývoja plodu. V tomto štádiu sa dieťa začína aktívne pohybovať a informácie o pozícii jeho tela začínajú putovať nervovým systémom, čo môže zároveň spôsobovať postupné vytváranie reprezentácií tela v mozgu. Podľa výskumu [4], kde skúmali reakcie mozgu na pasívny pohyb končatín u novorodených makakov, však po narodení reprezentácie tela v somatosenzorickom kortexe ešte nie sú vyvinuté. Ďalší vývoj teda prebieha postupne v rannom detstve, ale môže pokračovať aj v puberte a v niektorých prípadoch až do dospelosti [5]. Aby sme mohli vytváranie reprezentácie simulovať na robotovi iCub, bude nutné vytvoriť dataset, ktorý vhodným spôsobom odráža pohyb dieťaťa.
- Posledným faktorom, ktorý je nutné v snahe o vytvorenie biologicky inšpirovanej reprezentácie zohľadniť, je lateralita. Centrálnym nervovým systémom putujú jednotlivé somatosenzorické informácie najčastejšie kanálmi, ktoré sa na viacerých miestach v mieche môžu krížiť. Mnohé z nich tak prechádzajú z ľavej časti tela na pravú a naopak. Propriocepcia nie je výnimkou a preto je väčšina informácií z ľavej časti tela v primárnom somatosenzorickom kortexe reprezentovaná v pravej mozgovej hemisfére a naopak. Výsledné mapy v Brodmannovej oblasti 3a teda reprezentujú jednotlivé polovice tela zvlášť.

Kapitola 4

Implementácia a testy

V tejto kapitole budem popisovať spôsob riešenia z hľadiska samotnej implementácie. Najskôr popíšem vstupné dáta, ktoré som získala od vedúceho práce M. Hoffmanna, a ktoré budem na testovanie a porovnávanie reprezentácií navrhnutých a popísaných v predchádzajúcej kapitole 3 používať. Ďalej popíšem spôsob predspracovania dát, algoritmus učenia a samotnú implementáciu v Matlabe. Najviac sa však budem venovať voľbe vhodnej reprezentácie vstupných dát, ktorej možnosti najskôr popíšem a následne vhodnosť takto zvolených reprezentácií otestujem.

4.1 Implementácia

Pri implementácii som sa opierala o prácu Z. Straku [2], na ktorú nadväzujem využitím SOM toolboxu a to jednak v pôvodnej a jednak v modifikovanej verzii [9] [2]. Pre testovanie navrhovaného spôsobu reprezentácie proprioceptívnych vstupov pomocou Kohonenových máp teda použijem MATLAB.

Pôvodné rozšírenie SOM toolbox bolo vyvinuté prevažne na Helsinky University of Technology pre zjednodušenie práce s rôznymi modifikáciami algoritmu SOM [9]. Keďže na vytváraní SOM toolboxu sa podieľalo väčšie množstvo spolupracovníkov, obsahuje množstvo funkcií, ktoré rôzne modifikujú a dopĺňajú základný algoritmus a taktiež množstvo voliteľných argumentov funkcií. Z celej ponúkanej funkcionality toolboxu budem však prevažne využívať len základné funkcie pre učenie pôvodných Kohonenových máp, ktoré porovnam s modifikovanou verzou toolboxu somtoolbox05m2809 Z. Straku, kde pre výpočet vzdialenosti používa namiesto Euklidovskej vzdialenosti skalárny súčin. Nižšie popíšem časti toolboxu, ktoré v práci využívam.

4.1.1 SOM toolbox

V SOM toolboxe je výsledná mapa implementovaná ako štruktúra `sMap`, ktorá obsahuje samotnú naučenú mapu `sMap.codebook`, štruktúru topologické usporiadanie `sMap.topol` s typom usporiadania a veľkosťou výstupnej mapy a ďalšie parametre ako históriu učenia, či názvy komponentov, s ktorými v našom prípade nebudeme pracovať.

Funkciou, ktorá inicializuje výslednú mapu náhodnými hodnotami z rozsahu príslušných vstupných dát je `som_randinit`. Základnou funkciou toolboxu, v ktorej prebieha samotné učenie, je `som_seqtrain`. Na jej vstupe je uvedená štruktúra `sMap`, matica tréningových dát `D`, ktorá v riadkoch obsahuje jednotlivé vektory vstupných dát a ďalšie nepovinné argumenty. Pre účely tejto práce z nich budeme používať:

- `'sample_order' = 'random'/'ordered'` Defaultne sú vstupné dáta z tréningovej množiny vyberané v náhodnom poradí. Keďže pohyb, ktorý vytvára reprezentáciu propriocepcie u človeka je plynulý, pre poradie vyberania dát z tréningovej množiny môže byť vhodnejšie zvoliť usporiadaný výber, teda výber dát v takom poradí, v akom postupne vznikali.
- `'trainlen'` Týmto parametrom je upravovaný počet tréningových cyklov.

- ‘radius_ini’ počiatočný polomer funkcie okolia
- ‘radius_fin’ konečný polomer funkcie okolia

4.1.2 Vstupné dáta

Vstupné dáta pre učenie SOM zozbieral vedúci práce M. Hoffmann. Keďže u človeka vytváranie reprezentácie proprioceptívnych vstupov prebieha najmä u novorodencov a v rannom veku dieťaťa, pre generovanie relevantných dát sme zvolili pohyby robota, ktoré správanie takéhoto dieťaťa simulujú. iCub teda pohybuje rukami pred tvárou a tento pohyb sleduje očami, pričom zároveň otáča hlavou. Pre vytvorenie korektnej reprezentácie je zároveň nutné dodržať lateralitu a teda aj vstupné dáta zbierať tak, aby bolo možné mapy naučiť zvlášť pre ľavú a pravú časť tela. To znamená, že najskôr robot hýbe jednou rukou, pričom očami a hlavou tento pohyb sleduje, a až následne hýbe a sleduje pohyb druhej ruky.

Zozbierané dáta sú na priloženom CD v zložke `data` a podzložke `left_arm_and_head`. Pre načítanie dát do štruktúry `data` je možné použiť priložený script `data_read`, kde sú zároveň vydefinované rozsahy jednotlivých stupňov voľnosti v štruktúre `ranges`. Dáta sú zbierané s približnou frekvenciou 50Hz, čo je zhruba každých 20ms. Presné časové údaje vzniku dát sú uvedené v jednotlivých datasetoch vždy v prvých dvoch stĺpcoch v ms. Keďže táto frekvencia nie je presne zachovaná a zosúladená pre zber dát z rôznych častí tela, je nutné dáta nejakým spôsobom synchronizovať. Pre naše účely postačuje lineárna interpolácia, ktorá dáta zjednotí na frekvencii presne 50Hz. To znamená, že v každom čase $t_n \in [t_1 : 0.02 : t_f]$ nájdeme v množine vstupných dát také, ktorého príslušný čas t_0 je najväčší a zároveň menší ako t_n a také, ktorého čas vzniku t_1 je najmenší, ale zároveň väčší ako t_n . Interpolovanú j -tú hodnotu vektora d_n potom vypočítame ako:

$$d_{n,j} = d_{0,j} + (t_n - t_0) \frac{d_{1,j} - d_{0,j}}{t_1 - t_0}$$

Bližšie je synchronizácia popísaná priamo v scripte `data_synchro_left` v zdrojovom kóde.

S dátami po načítaní pracujem ako so štruktúrou `joints`, ktorá obsahuje jednotlivé časti tela: `joints.head`, `joints.left_arm`, `joints.right_arm`, `joints.torso`. V prípade datasetu `left_arm_and_head`, ktorý využívam v tejto práci, sú to hlavne `joints.head` a `joints.left_arm`.

4.2 Predspracovanie dát

Pred samotným testovaním navrhovanej reprezentácie je nutné otestovať parametre a správanie populačného kódovania v algoritme SOM, preto na začiatok popíšem zakódovanie a učenie pre 1 stupeň voľnosti a následne budem reprezentáciu testovať na celej množine vstupných dát. Parametre, ktoré je možné upravovať a tým výrazne meniť a ovplyvniť reprezentáciu sú najmä:

- Parametre kódovania – u sigmoidy je to odsadenie a sklon, u Gaussových kriviek je to stredná hodnota a rozptyl.
- Počet neurónov v populácii, ktorými je zakódovaný 1 stupeň voľnosti.
- Spôsob výpočtu vzdialenosti vstupných dát od jednotlivých vektorov váh neurónov SOM – najmä porovnanie Euklidovskej vzdialenosti a skalárneho súčinu.
- Sample order – parameter učenia SOM, ktorý reprezentuje poradie vyberania vektorov z množiny vstupných dát.

- Počet opakovaní a radius – parametre učenia SOM.

Pri každom z testov popíšem nastavenie parametrov učenia, typ testovanej reprezentácie a názov scriptu, ktorým je možné daný test spustiť a vykresliť ilustratívne grafy. Pre hodnotenie reprezentácií je tiež dôležité zistiť čo výsledné dáta zakódované populáciou neurónov reprezentujú a teda pokúsiť sa dáta spätne pretransformovať a určiť tak uhol, prípadne rozsah stupňa voľnosti, ktorý daný neurón výslednej SOM reprezentuje.

Spoločné nastavenia parametrov učenia budem pri testoch popisovať nasledovne:

- `som_size` označuje veľkosť výslednej mapy SOM, napríklad `[7 7]` pre mapu 7x7.
- `number_of_dof` je počet stupňov voľnosti, pre ktoré budem v danom teste vytvárať reprezentáciu.
- Parameter `input` uchováva maticu vybraných vstupných dát pre daný test, napríklad `[joints.head joints.left_arm]` pre testy kombinujúce vstupné dáta z hlavy a ľavej ruky, alebo napr. `[joints.head(1:200,1)]` pre prvých 200 vstupných dát prvého stupňa voľnosti hlavy.
- V zdrojovom kóde figuruje v nastaveniach zároveň parameter `input_ranges`, ktorý obsahuje rozsahy vybraných stupňov voľnosti v `input`, ale tento parameter vychádza z predchádzajúceho, preto ho pre prehľadnosť nebudem v texte uvádzať.
- Napokon vždy uvediem o aké populačné kódovanie sa jedná, jeho parametre a tiež testované parametre učenia SOM ako spomínaný `som_order`, `som_trainlen`, alebo `som_radius_ini` a `som_radius_fin`. Ak niektoré z týchto parametrov neuvediem, nastavenie bolo ponechané defaulté – viď [9].

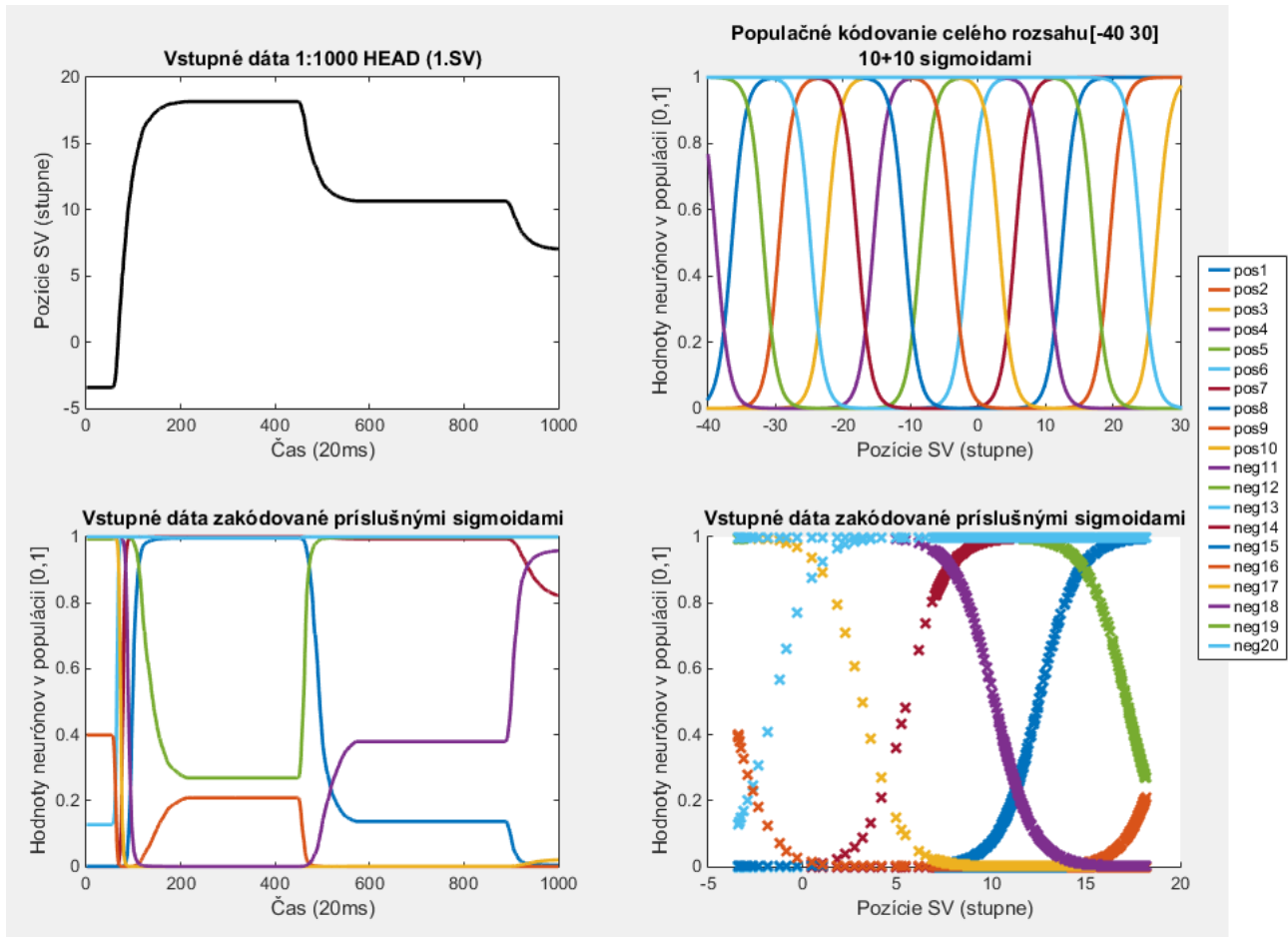
Populačné kódovanie pri väčšom počte stupňov voľnosti s rôznymi rozsahmi môže byť dvojakého charakteru. Jednak je možné stanoviť konštantný počet neurónov, ktoré budú kódovať každý SV, s tým, že parametre sa prepočítajú podľa príslušného rozsahu, alebo je možné zvoliť priamo konštantné parametre jednotlivých funkcií a prepočítať počet neurónov, ktorý sa bude pre jednotlivé SV líšiť. Podľa toho ktorú možnosť zvolíme, vyberieme si funkciu na prepočítanie parametrov končiacu na `_var` pre variabilné parametre s konštantným počtom neurónov, alebo `_cons` pre konštantné parametre. Na výstupe týchto funkcií je vždy štruktúra `params`, ktorá obsahuje konkrétne parametre zvoleného kódovania. Spoločným parametrom pre všetky kódovania je `params.counter`, vektor, ktorého zložky reprezentujú počty neurónov kódujúce jednotlivé stupne voľnosti.

4.2.1 Sigmoidy

Parametre, ktorými budem ďalej v testoch označovať, že sa jedná o populačné kódovanie sigmoidami, sú:

- `number_of_sig`, ktorý označuje počet pozitívne, alebo negatívne orientovaných sigmoid, tzn. vo finále je 1 SV zakódovaný počtom $2 \times \text{number_of_sig}$ sigmoid, v prípade, že chceme zakódovať všetky SV rovnakým konštantným počtom neurónov.
- `offset` označuje odstupy jednotlivých neurónov v populácii v prípade, že chceme zachovať rovnaké odstupy a počet neurónov kódujúcich jednotlivé SV sa môže líšiť.

Pre populačné kódovanie sigmoidami môžeme použiť najskôr funkciu pre prepočítanie parametrov kódovania `params.sig_var`, alebo `params.sig_cons` a následne `sig_population` pre zakódovanie vstupu populáciou neurónov s danými parametrami. Ukážka ako sa dajú vstupné dáta `input = joints.head(1:1000,1)` zakódovať sigmoidami, je napr. obrázok 4.1.



Obrázok 4.1. Príklad zakódovania vstupných dát populáciou 10+10 sigmoíd. Graf vľavo hore vykresľuje hodnoty vstupných dát v stupňoch vzhľadom na čas, kedy vznikli. Vykreslený pohyb teda označuje rotáciu v jednom SV. Vpravo hore je populácia vstupných neurónov, ktoré 10 + 10 sigmoídami kódujú celý potencionálny rozsah daného SV. Na spodných grafoch je znázornené zakódovanie vstupných dát v input touto populáciou. Vľavo môžeme vidieť kódovanie 'v čase' a krížiky vpravo označujú ako sú konkrétne vstupy v uhloch zakódované hodnotami príslušných sigmoíd. To znamená, že jeden vstup je kódovaný všetkými 'krížikmi', ktoré sa v príslušnej hodnote uhla nachádzajú vertikálne nad sebou.

Pre hodnotenie takto reprezentovaných vstupov budem používať spätnú transformáciu:

$$y = \frac{1}{1 + e^{\text{sgn}(-x + \text{offset})}}$$

$$e^{\text{sgn}(-x + \text{offset})} = \frac{1 - y}{y}$$

$$x = -\frac{\ln(1 - y) - \ln(y) - \text{offset} \cdot \text{sgn}}{\text{sgn}}$$

Pre vysvetlivky viď kapitola 3. V zdrojovom kóde je spätná transformácia implementovaná vo funkcii `sig_decode`.

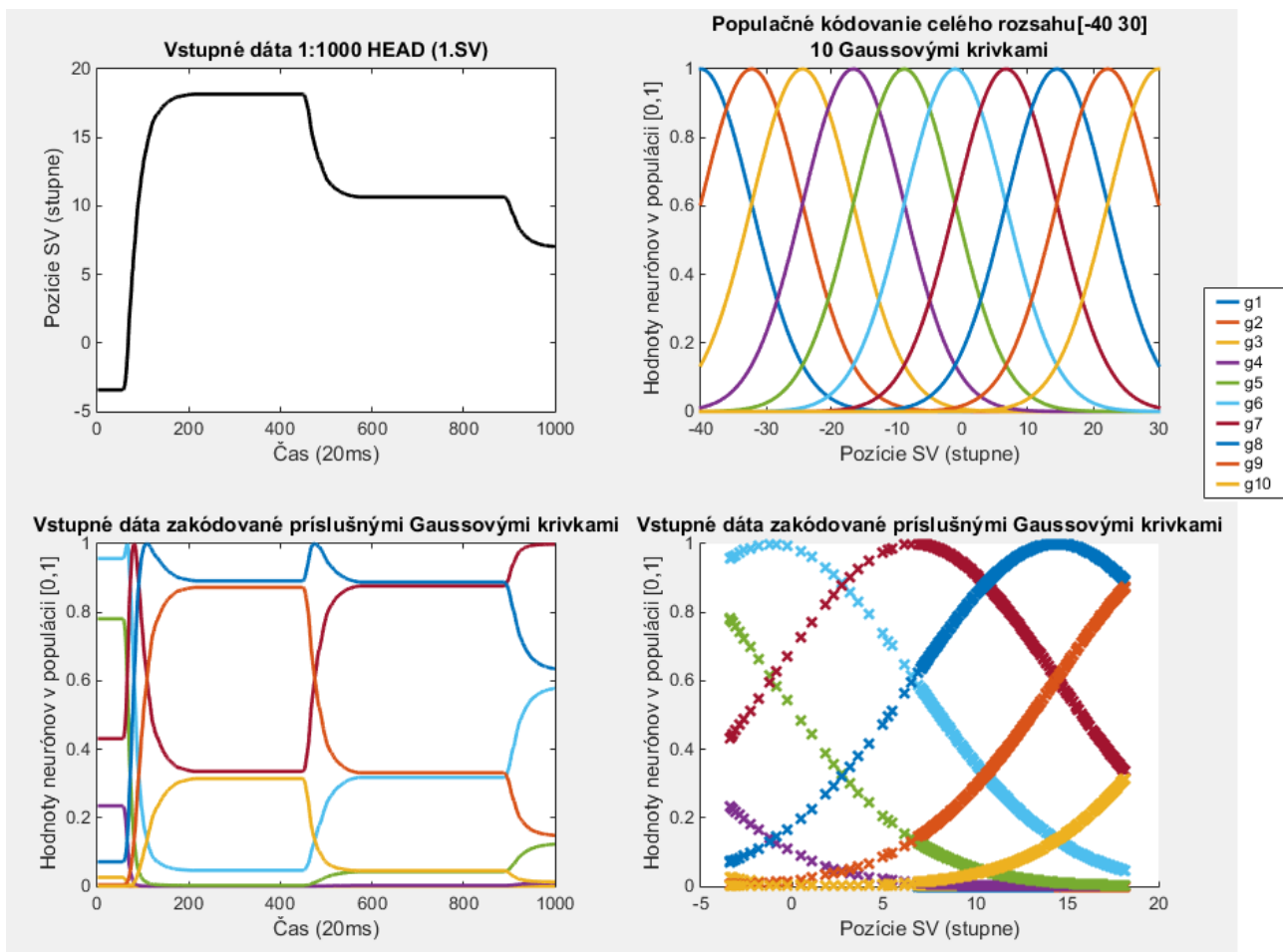
4.2.2 Gaussove krivky

Parametre, ktorým budem naopak označovať kódovanie Gaussovými krivkami, sú:

- `number_of_gauss` označuje počet Gaussových kriviek, ktorými je zakódovaný každý stupeň volnosti.
- `sigma` označuje rozptyl Gaussovej krivky sigma v prípade, že chceme, aby bol tento parameter spoločný pre všetky SV.
- `offset` podobne ako u sigmoid označuje odstupy jednotlivých Gaussových kriviek v populácii, respektíve ich stredných hodnôt.

Pre gaussovské populačné kódovanie môžeme použiť najskôr funkciu pre prepočítanie parametrov kódovania `params_gauss_var`, alebo `params_gauss_cons` a následne `gauss_population` pre zakódovanie vstupu populáciou neurónov s danými parametrami.

Ukážka zakódovania vstupných dát populáciou Gaussových kriviek na obrázku 4.2:



Obrázok 4.2. Príklad zakódovania vstupných dát populáciou 10 Gaussových kriviek. Pre vysvetlivky viď obrázok 4.1.

Pre hodnotenie a zobrazenie je tiež použitá spätná transformácia a to:

$$y = e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

$$\ln(y) = \frac{-(x-\mu)^2}{2\sigma^2}$$

$$(x - \mu)^2 = -\ln(y) \cdot 2\sigma^2$$

$$x = \mu + \sqrt{-\ln(y) \cdot 2\sigma^2}$$

Odmocnina, ktorú po úprave dostaneme, znamená, že každá výsledná hodnota jedného neurónu označuje/ kóduje dve rôzne hodnoty uhlu. Z nich môžeme vždy vylúčiť tie, ktoré spadajú mimo rozsah daného stupňa voľnosti a tiež tie, ktoré spadajú mimo rozsah vstupných dát (vstupné dáta totiž nemusia pokrývať celé rozsahy jednotlivých stupňov voľnosti). Dekódovanie populačného kódovania Gaussovými krivkami je implementované vo funkcii `gauss.decode`.

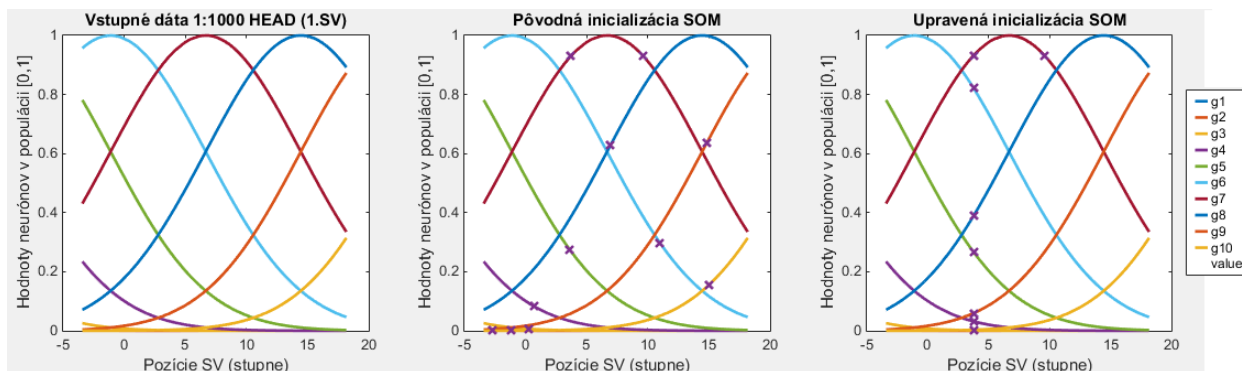
4.2.3 Lineárne funkcie

Po naimplementovaní zakódovania a spätnej transformácie sa ukázalo populačné kódovanie lineárnymi funkciami ako nevhodné pre naše účely, pretože každá z lineárnych funkcií je na určitom intervale (mimo oboru hodnôt v intervale $[0,1]$) konštantne vynulovaná, čo znásobuje zložitosť spätnej transformácie kódovaného uhla, nakoľko intervaly, na ktorých sú jednotlivé funkcie nulové, sú vzhľadom na celý rozsah daného SV pomerne veľké. Dochádza teda rýchlejšie k nekonzistencii dát (viď časť 4.3). Pre bližšie info viď zdrojový kód - ukážka `test.lin.encode`.

4.3 Problémy populačného kódovania

Teraz, keď som popísala ako sa dá uhol spätne pretransformovať, popíšem na príklade gaussovského populačného kódovania pre jeden stupeň voľnosti, prečo je táto spätná transformácia problematická.

Majme vstupné dáta z datasetu `left_arm_and_head`, z ktorých použijeme iba rotáciu prvého stupňa voľnosti hlavy (`neck pitch` – viď príloha B) s rozsahom $[-40,30]$, teda `input = [joints.head(0:1000,1)]`. Pre zjednodušenie zvolíme veľkosť výslednej mapy SOM `som_size = [1 1]` a definujeme počet Gaussových kriviek, ktoré budú kódovať daný stupeň voľnosti `number_of_gauss = 10` rovnako ako v obrázku 4.2. Po zakódovaní vstupných dát dostaneme populáciu 10 neurónov, zobrazenú na obrázku 4.3 vľavo. Následne chceme náhodne nainicializovať výslednú mapu $[1 1]$. Funkcia pre takúto inicializáciu je v SOM toolboxe naimplementovaná ako `som_randinit` a funguje tak, že sa pre každý prvok vektoru váh jednotlivých neurónov pozrie na rozsah príslušných vstupných dát a priradí mu náhodnú hodnotu z tohoto rozsahu. To znamená, že každému prvku, ktorý odpovedá jednému vstupnému neurónu v populácii, priradí hodnotu z intervalu $[0,1]$, čím po spätnej transformácii dostávame 10×2 hodnoty pozície SV v stupňoch, ktoré spolu vo výsledku nepredstavujú žiaden uhol, ako je patrné zo stredného grafu v obrázku 4.3. Preto som inicializáciu upravila vo funkcii `som_randinit_gauss`, respektíve `som_randinit_sig` tak, aby na jej vstupe boli pôvodné dáta v stupňoch. Najskôr je teda náhodne zvolená hodnota z rozsahu príslušných vstupných dát v stupňoch a až tá je následne zakódovaná populáciou neurónov. Po spätnej transformácii teda dostaneme konkrétny uhol, viď obrázok 4.3 vpravo.



Obrázok 4.3. Ukážka pôvodnej a upravenej inicializácie SOM

Ani to však ešte nezabezpečí, aby výsledná mapa algoritmu SOM po spätnej transformácii prvkov kódujúcich jednotlivé stupne voľnosti reprezentovala presne, alebo aspoň približne, určitú pozíciu príslušného stupňa voľnosti. Funkcia vzdialenosti v každom kroku ku každému vektoru váh mapy pripočíta rozdiel medzi ním a vektorom d_i na vstupe, prenasobený príslušnými koeficientami vzdialenosti. Majme napríklad rovnaké parametre ako v predchádzajúcom prípade, okrem `input = joints.head(:,1)`, a na vstupe vektor d_i , zakreslený na obrázku 4.4 červenou farbou, ktorý kóduje 10tismi Gaussovými funkciami uhol približne 3.755° .

$$d_i = [0.0000 \ 0.0000 \ 0.0014 \ 0.0318 \ 0.2667 \ 0.8222 \ 0.9324 \ 0.3889 \ 0.0597 \ 0.0034]$$

A majme výslednú mapu [1 1] s jedným neurónom, ktorého vektor váh m_c je inicializovaný na -11.788° (na obrázku modrou farbou):

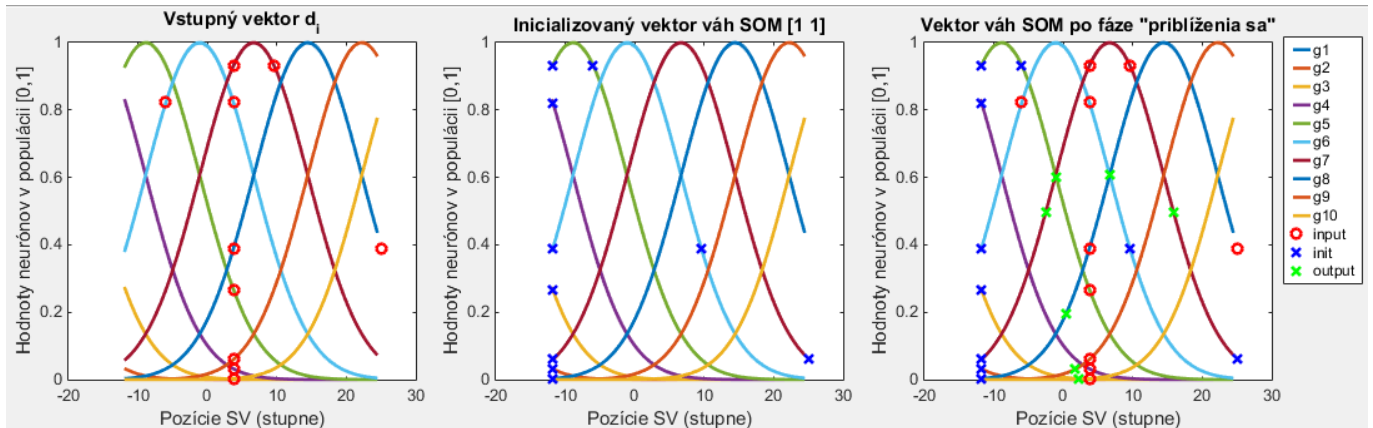
$$m_c(t) = [0.0014 \ 0.0317 \ 0.2661 \ 0.8214 \ 0.9329 \ 0.3897 \ 0.0599 \ 0.0034 \ 0.0001 \ 0.0000]$$

Podľa funkcie vzdialenosti bude potom výsledný vektor $m_c(t+1)$:

$$m_c(t+1) = m_c(t) + \alpha(t)h_{cc}(t)[d_i(t) - m_c(t)]$$

$$m_c(t+1) = [0.0007 \ 0.0159 \ 0.1337 \ 0.4266 \ 0.5998 \ 0.6060 \ 0.4961 \ 0.1962 \ 0.0299 \ 0.0017]$$

Na obrázku je tento vektor po spätnej transformácii zakreslený zelenou farbou. Z toho istého obrázku je zároveň patrné prečo je spätná transformácia populačného kódovania problematická. Keďže krivky majú na rôznych miestach rôzny sklon, niektoré prvky vektorov váh sa budú 'približovať' pomalšie ako iné a dáta tak prestanú byť konzistentné. Ďalším problémom je, že každá Gaussova krivka dá po spätnej transformácii dve rôzne hodnoty uhlov. Pri spätnej transformácii konkrétneho uhla je výslednou hodnotou tá, ktorá je spoločná pre všetky krivky. Ak chceme ale odhadnúť uhol, respektíve rozsah, ktorý je zakódovaný populáciou neurónov po naučení, musíme dáta nejakým spôsobom vyhodnotiť/aproximovať.

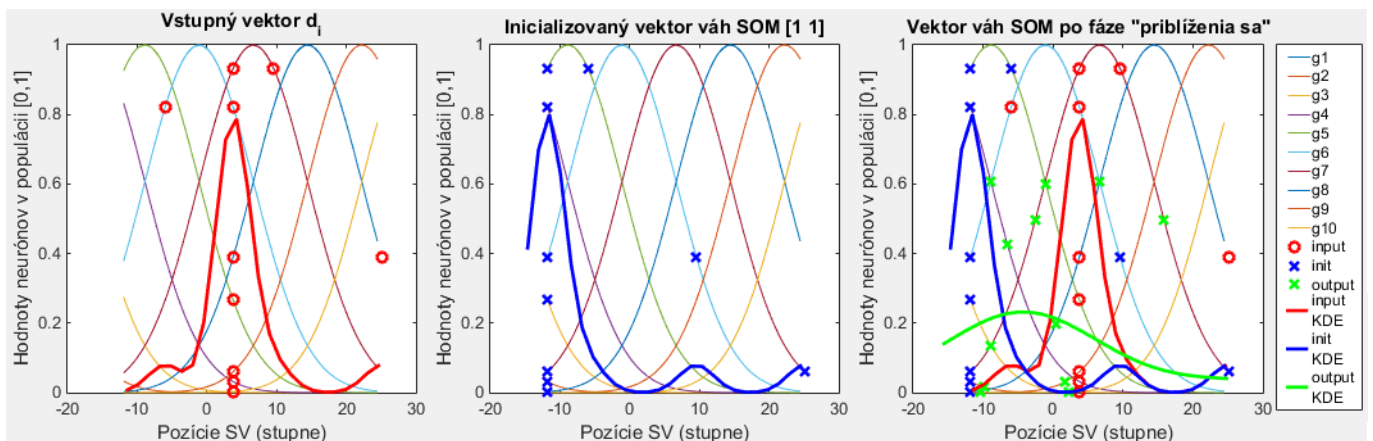


Obrázok 4.4. Ukážka funkcie 'približovania sa'.

Najviac sa osvedčila neparametrická metóda odhadu Kernel Density Estimation ¹⁾, ktorá aproximuje funkciu hustoty pravdepodobnosti vstupných dát pomocou funkcie:

$$f_h(x) = \frac{1}{nh} \sum_{i=1}^N K\left(\frac{x - x_i}{h}\right); \quad -\infty < x < \infty$$

Kde h je takzvaný 'bandwidth', šírka okna, ktorá ovplyvňuje 'hladkosť' odhadu, K_h je normalizovaná pravdepodobnostná funkcia, napríklad Gaussova a $x_i, i = 1..n$ sú vstupné dáta. Keďže algoritmus KDE je implementovaný ako súčasť Matlabu, pre vyhodnotenie som používala priamo funkciu Matlabu `ksdensity`. Obrázok 4.5 je ukážkou vyhodnotenia spätnej transformácie populačného kódovania z obrázku 4.4:



Obrázok 4.5. Ukážka transformácie pomocou KDE.

Globálne maximum v prípade KDE spätnej transformácie nainicializovaného vektora váh SOM dá -11.5203° pri zachovaní defaultného bandwidth funkcie `ksdensity` v Matlabe, čo s odchýlkou odpovedá pôvodným -11.788° , alebo pri znížení bandwidth na 0.2 dostaneme -11.7612 , čo je oveľa presnejší odhad, ale výsledná krivka bude mať ostrejšie a výraznejšie hrany. Pre vstupný vektor dostaneme odhad 4.0726° . Na zelenej krivke na obrázku vpravo môžeme vidieť, že neoznačuje s takou presnosťou jeden konkrétny uhol, ale skôr okolie v rozmedzí zhruba od -20 do 10 so stredom približne v -5.0543° .

¹⁾ <http://www.mathworks.com/help/stats/kernel-distribution.html>

4.4 Algoritmus

Z dôvodu využitia populačného kódovania je používaný postup oproti pôvodnému algoritmu SOM popísanému v kapitole 3 mierne modifikovaný. Uvádžam preto pre prehľadnenie celý postup, ktorým budem dáta od načítania až po vyhodnotenie finálnej mapy spracovávať.

1. Načítanie dát
2. Synchronizácia dát lineárnou interpoláciou
3. Zakódovanie vstupov populáciami neurónov
4. Inicializácia výslednej mapy SOM a následné zakódovanie inicializovanej mapy populáciami neurónov s rovnakými parametrami ako v predchádzajúcom kroku
5. Samotné učenie SOM bez modifikácie
6. Spätná transformácia/dekódovanie výstupnej mapy pomocou Kernel Density Estimation
7. Vyhodnotenie výslednej mapy

4.5 Testovanie

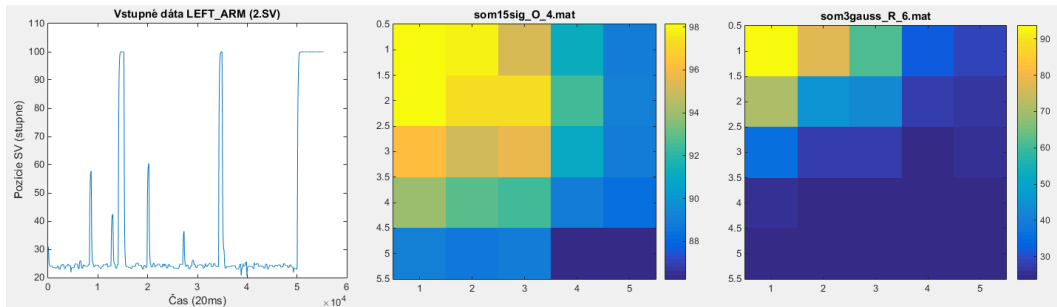
Teraz, keď som popísala rôzne spôsoby zakódovania a spätnej transformácie vstupov, budem spomínané parametre a ich vplyv na výslednú mapu testovať. Postupne popíšem kritériá, ktoré by mali výsledné mapy v našom prípade spĺňať, spôsob, akým je možné tieto kritériá hodnotiť a pre každé kritérium uvediem z vytvorených máp príklad takej, ktorá ho spĺňa a takej, ktorá ho nespĺňa. Zároveň budem uvedené vlastnosti ilustrovať obrázkami. Pre celý zoznam testovaných konfigurácií a ich ohodnotenie viď priložený súbor `testy.xls`.

Výsledná mapa by mala spĺňať nasledujúce požiadavky:

- **Reprezentácia vstupných dát** – Typickou vlastnosťou samoorganizujúcich sa máp je, že váhové vektory neurónov výslednej mapy priamo reprezentujú charakteristické hodnoty, vlastnosti, alebo parametre vstupných dát, pričom správne naučená mapa by mala pokrývať čo najväčší rozsah vstupných dát cez všetky zložky jej vektorov. Túto vlastnosť je v našom prípade možné overiť pre jednotlivé SV vo výslednej mape napríklad vykreslením jej prierezu cez tieto zložky, respektíve cez hodnoty uhlov, ktoré príslušné prvky jednotlivých vektorov váh výslednej mapy kódujú. Majme napríklad mapu `test03/som15sig_0.4.mat` s nasledujúcimi parametrami:


```
input = [joints.head joints.left_arm]; number_of_dof = 22;
som_size = [5 5]; som_order = 'ordered'; trainlen = 4;
number_of_sig = 15;
```

Výsledná mapa obsahuje 5×5 neurónov. Každý z neurónov reprezentuje 22 stupňov voľnosti, z ktorých každý je zakódovaný 15+15 sigmoidami, teda vektor váh každého neurónu obsahuje 660 prvkov. Ak sa pozrieme na prierez takouto mapou a vezmeme z každého neurónu napríklad 30 prvkov odpovedajúcich 8. stupňu voľnosti, teda prvky 211 až 240, tie budú po spätnej transformácii popísanej v časti 4.3 kódovať 5×5 uhlov, resp. rozsahov. Rozsah vstupných dát 8. SV z `input`, resp. 2. SV ľavej ruky, je [20.6021, 100.0314], viď obrázok 4.6 vľavo. Rozsah, ktorý uvedená naučená SOM pre tento SV kóduje je [86.2499, 98.1270] – viď obrázok 4.6 v strede, čo zďaleka nepokrýva rozsah vstupných dát.



Obrázok 4.6. Ukážka miery reprezentácie vstupných dát výslednou mapou po spätnom dekódovaní pomocou KDE pre 2. SV ľavej ruky.

Mapa s takýmito parametrami teda nespĺňa požiadavok na reprezentáciu celého rozsahu vstupných dát. Pre porovnanie majme mapu `test01/som3gauss_R.6.mat` s parametrami:

```
input = [joints.head joints.left_arm]; number_of_dof = 22;
som_size = [5 5]; som_order = 'random'; trainlen = 6;
number_of_gauss = 3;
```

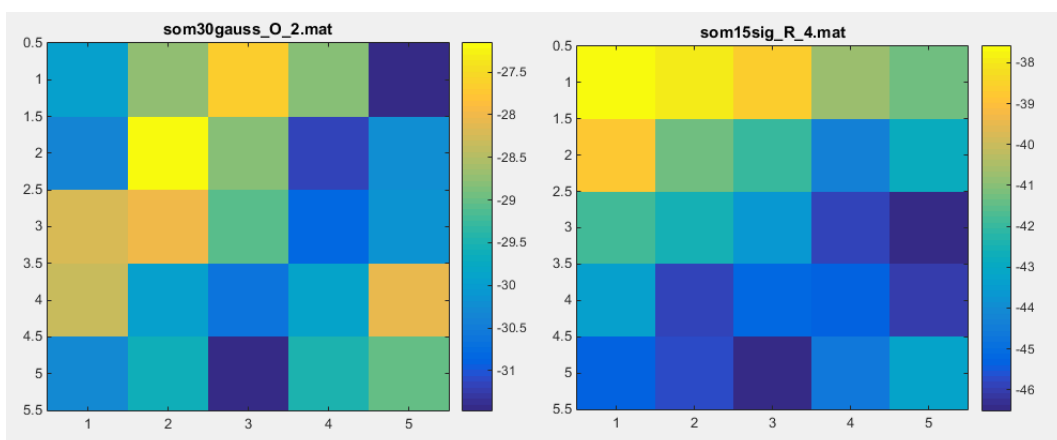
Z obrázku 4.6 vpravo je naopak vidieť, že táto mapa pre daný stupeň voľnosti uvedený požiadavok spĺňa. Dekódované hodnoty pokrývajú rozsah [23.7933, 93.6528], čo je približne celý rozsah daného SV.

- Zachovanie topológie** – Ako bolo spomenuté v kapitole 3, samoorganizujúce sa mapy tiež zachovávajú topologické usporiadanie vstupných dát, teda podobné vstupné dáta sú v mape reprezentované blízko seba. Táto vlastnosť sa dá jednoducho overiť napríklad po vykreslení dekódovaného prierezu výslednej mapy cez jednotlivé stupne voľnosti, podobne ako v predchádzajúcom kroku. Vlastnosť zachovania topológie je možné porovnať na obrázku 4.7 pre mapu `test01/som30gauss_0.2.mat` (na obrázku vľavo) s parametrami:

```
input = [joints.head joints.left_arm]; number_of_dof = 22;
som_size = [5 5]; som_order = 'ordered'; trainlen = 2;
number_of_gauss = 30;
```

A mapu `test04/som15sig_R.4.mat` (na obrázku vpravo) s parametrami:

```
input = [joints.head joints.left_arm]; number_of_dof = 22;
som_size = [5 5]; som_order = 'random'; trainlen = 4; offset = 15;
```

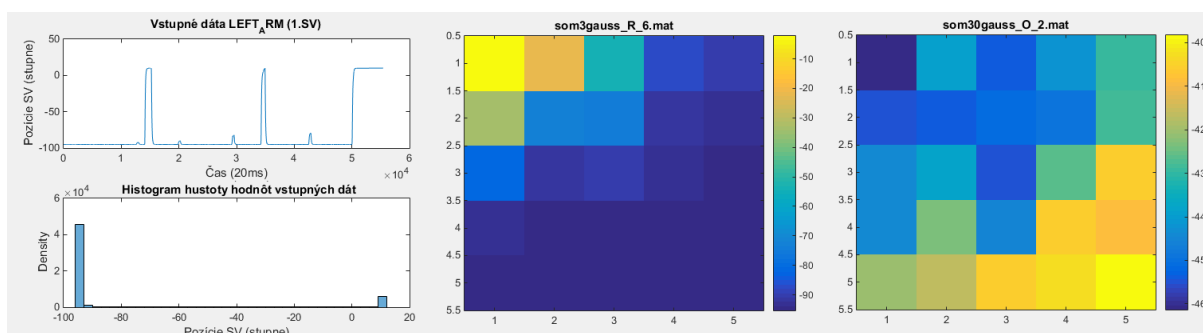


Obrázok 4.7. Ukážka topografického usporiadania výslednej mapy [5 5] pre 12. stupeň voľnosti vstupných dát, resp. 6. SV ľavej ruky.

- Pomerné zastúpenie vstupných dát** – Ďalšou z charakteristických vlastností výsledných máp architektúry SOM je zachovávanie pomerného zastúpenia hodnôt vstupných dát, čo znamená, že pomer neurónov mapy kódujúcich určitú hodnotu odráža pomer týchto hodnôt v množine vstupných dát. Táto vlastnosť je zobrazená na obrázku 4.8, kde je najskôr vľavo vykreslený priebeh vstupných dát pre 1. SV ľavej ruky spolu s histogramom, ktorý reprezentuje hustotu týchto hodnôt na vstupe. V strede na obrázku sa nachádza mapa `test01/som3gauss_R_6.mat` s parametrami:

```
input = [joints.head joints.left_arm]; number_of_dof = 22;
som_size = [5 5]; som_order = 'random'; trainlen = 6;
number_of_gauss = 3;
```

Ako je vidieť z colorbaru a histogramu, táto mapa pomerne dobre odráža hustotu hodnôt vstupných dát - väčšia časť mapy reprezentuje hodnoty okolo -100° , stredné hodnoty sú pokryté minimálne a zvyšok mapy predstavuje hodnoty okolo 15° , čo odpovedá hustote týchto hodnôt na vstupe. Naopak celá mapa na obrázku vpravo kóduje hodnoty od -40° do -46° , pričom tieto hodnoty sú v množine vstupných dát zastúpené minimálne.



Obrázok 4.8. Ukážka pomerného zastúpenia vstupných hodnôt vo výslednej mape [5 5] pre 8. stupeň voľnosti vstupných dát, resp. 1. SV ľavej ruky.

Keďže pri väčšom množstve stupňov voľnosti je vyššie popísaný spôsob hodnotenia pomerne zložitý, kvalita výslednej reprezentácie sa dá určiť aj pomocou matematického výpočtu. Pre hodnotenie výsledných máp algoritmu SOM sa najčastejšie používajú nasledujúce odhady:

- Topologická chyba** – Topologická chyba je počítaná ako pomer všetkých vstupných vektorov, pre ktoré prvý a druhý najbližší neurón (BMU) vo výslednej mape nie sú susedné.
- Kvantizačná chyba** – Ak vlastnosť reprezentácie vstupných dát a ich pomerného zastúpenia vo výslednej mape zovšeobecníme na všetky stupne voľnosti, môžeme kvalitu reprezentácie odhadovať výpočtom takzvanej kvantizačnej chyby. Táto hodnota označuje priemernú vzdialenosť všetkých vstupných dát od ich príslušných ‘najbližších’ neurónov (BMU) výslednej mapy.

Výpočet kvantizačnej qe aj topologickej chyby te je implementovaný ako súčasť SOM toolboxu vo funkcii `[qe te] = som_quality()`. Ich hodnoty pre jednotlivé testované mapy je možné nájsť v priloženom súbore `testy.xls`

Z množstva konfigurácií, ktoré boli v rámci tejto práce otestované, je možné na základe kritérií popísaných vyššie určiť ktoré parametre populačného kódovania a učenia prispievajú ku kvalitnejšej reprezentácii. Ako môžeme vidieť v priloženom dokumente `testy.xls`, najlepšie výsledky dáva gaussovské populačné kódovanie s konštantným počtom neurónov v populácii. Zároveň čím menší počet neurónov daný kĺb kóduje, tým

je výsledná mapa presnejšia, čo môže súvisieť aj s primeraným zvyšovaním parametru sigma a teda znižovaním rýchlosti rastu/klesania funkcie, takže fáza ‘približovania sa’ počas učenia SOM je konzistentnejšia, čo sa týka kódovania konkrétneho uhlu. Tiež sa osvedčilo ako vhodnejšie zvoliť spôsob výberu vstupných dát náhodne a zvýšiť počet tréningových cyklov.

Mapa s najmenšou topologickou a kvantizačnou chybou a zároveň mapa, ktorá najlepšie spĺňala požadované vlastnosti, je `test01/som3gauss_R.6.mat`. Keďže vizualizácia tak veľkého počtu stupňov voľnosti je pomerne zložitá, na záver pripájam ukážku vizualizácie všetkých stupňov voľnosti tejto mapy priamo na postavení príslušných častí tela robota iCub. Na obrázku 4.9 je vidieť, že aj celková mapa zachováva topológiu a pomerne dobre reprezentuje proprioceptívne vstupy z množiny vstupných dát.



Obrázok 4.9. Ukážka výslednej mapy s populačným kódovaním, po spätnom dekódovaní pomocou KDE. Kamera má statickú pozíciu.

Kapitola 5

Záver

Cieľom tejto práce bolo vytvoriť reprezentáciu propioceptívnych vstupov humanoidného robota iCub, inšpirovanú reprezentáciou týchto vstupov u človeka. Počas dôkladného štúdia príslušných častí neurovedy sa ukázalo, že propiocepcia, jej kódovanie a spôsob reprezentácie, doposiaľ nie sú presne detailne zmapované a popísané. Experimenty a štúdiá dospievajú často k protichodným záverom a materiály a postupy, ktoré v pokusoch používajú nie sú vždy dôveryhodné, či podložené. Napriek tomu existujú všeobecne uznávané prístupy a závery, ktoré sme v rámci tejto práce pomerne komplexne zhrnuli a na ktoré sme následne naviazali modelovaním týchto procesov u robota iCub. Práca je teda pomerne uceleným rozborom nastolenej problematiky s načrtnutými spôsobmi riešenia a zhrnutými možnosťami a smermi, kadiaľ by bolo možné uberať sa ďalej, spolu s problémami, ktoré je treba do budúcnosti pri vytváraní biologicky inšpirovanej reprezentácie propiocepcie riešiť.

5.1 Diskusia

Riešenia a postupy v tejto práci vychádzajú prevažne z výskumov uvedených v Literatúre. Prístup, ktorý sme zvolili však zďaleka nie je jedinou možnosťou ako sa k danej problematike postaviť. V prvom rade je možné zvoliť rôzne datasety, ktoré vedú k rozdielnym výsledným mapám. Cieľom tejto práce nebolo vytvoriť presnú a komplexnú mapu, ktorá bude reprezentovať celý pohybový rozsah a propiocepciu robota iCub, ale skôr navrhnúť a otestovať spôsob akým by mohla byť reprezentácia vytvorená, prepojená so vstupnou vrstvou a interpretovaná. Dataset, ktorý sme zvolili, tieto účely spĺňa. Pre vytvorenie komplexnejšej reprezentácie však bude nutné zvoliť rozsiahlejšie pohyby rôzneho charakteru.

Hoci v rámci práce sme sa snažili pokryť všetky najpoužívanejšie spôsoby populačného kódovania, je možné použiť aj iné typy funkcií vstupných neurónov, napríklad binarizovanú verziu Gaussovského populačného kódovania [24], alebo rôzne upraviť verzie implementované v tejto práci. Ako som popísala v kapitole 4, pri použití populačného kódovania dochádza počas učenia SOM k narúšaniu konzistencie kódovaných dát. Túto vlastnosť by eventuálne tiež bolo možné korigovať, napríklad prenasobovaním prvkov vektorov zakódovaných vstupných dát vhodnými koeficientami, ktoré by spôsobili, že hodnoty na okrajoch intervalu (v našom prípade $[0,1]$) by sa približovali rýchlejšie, čím by ich miera 'približovania sa' približne vyrovnala s prvkami v strednom rozsahu hodnôt.

Poslednou fázou je samotné vyhodnocovanie naučenej mapy, pre ktoré sme zvolili možnosť spätnej transformácie populačného kódovania pomocou Kernel Density Estimation, čím dostaneme naspäť približné pozície jednotlivých stupňov voľnosti pre každý neurón vo výslednej mape v stupňoch. Takto transformované mapy zachovávajú požadované vlastnosti a pomerne dobre reprezentujú vstupné hodnoty. Je však samozrejme možné zvoliť iný prístup aj pre dekódovanie a vyhodnocovanie takto naučených máp.

Napriek rôznym problémom, ktoré v práci popisujem, sa nám podarilo vytvoriť reprezentácie, ktoré spĺňajú požadované kritériá a pomerne dobre odrážajú vstupné dáta. Po prehodnotení charakteru proprioceptívnych dát sme však dospeli k záveru, že bez manuálneho nastavenia receptívnych polí jednotlivých neurónov výslednej mapy [2] nie je kódovaním samotných pozícií kĺbov možné dospieť k reprezentácii podobnej Brodmanovej oblasti 3a v mozgu primátov. V snahe o čo najpresnejšie modelovanie tejto reprezentácie by teda stálo za bližšie štúdium prepojenie jednotlivých aferentných nervových ciest a rôznych oblastí v mozgu, čím by sme pri vytváraní požadovanej mapy získali ďalšie doplnkové informácie, napríklad o smere, či rýchlosti pohybu, alebo o pozícii takzvaného end-pointu, ako je bližšie popísané v kapitole 2.

Literatúra

- [1] L. N. De Castro. *Fundamentals of natural computing: basic concepts, algorithms, and applications*. CRC Press, 2006.
- [2] Z. Straka. *Tvorba senzorických map z taktilních vstupů robota iCub pomocí metod strojového učení*. Czech Technical University, Prague, Czech Republic, 2014.
- [3] E. R. Kandel, J. H. Schwartz, and T. M. Jessell. *Principles of Neural Science*. McGraw-Hill Medical, 4th edition, July 2000.
- [4] L. Krubitzer, K. J. Huffman, E. Disbrow, and G. Recanzone. Organization of area 3a in macaque monkeys: contributions to the cortical phenotype. *Journal of Comparative Neurology*, 471(1):97–111, 2004.
- [5] U. Proske and S. C. Gandevia. The proprioceptive senses: their roles in signaling body shape, body position and movement, and muscle force. *Physiological reviews*, 92(4):1651–1697, 2012.
- [6] D. Zipser and R. A. Andersen. A back-propagation programmed network that simulates response properties of a subset of posterior parietal neurons. *Nature*, 331(6158):679–684, 1988.
- [7] M. Bergenheim, E. Ribot-Ciscar, and J. P. Roll. Proprioceptive population coding of two-dimensional limb movements in humans: I. muscle spindle feedback during spatially oriented movements. *Experimental brain research*, 134(3):301–310, 2000.
- [8] V. G. Macefield. Physiological characteristics of low-threshold mechanoreceptors in joints, muscle and skin in human subjects. *Clinical and Experimental Pharmacology and Physiology*, 32(1-2):135–144, 2005.
- [9] J. Vesanto, J. Himberg, E. Alhoniemi, and J. Parhankangas. *SOM toolbox for Matlab 5*. Citeseer, 2000.
- [10] C. S. Sherrington. On the proprioceptive system, especially in its reflex aspect. *Brain*, 29:467–482, 1907.
- [11] R. Shadmehr and S. P. Wise. Supplementary documents for computational neurobiology of reaching and pointing, 2005.
- [12] G. Macefield, S. C. Gandevia, and D. Burke. Perceptual responses to microstimulation of single afferents innervating joints, muscles and skin of the human hand. *The Journal of Physiology*, 429(1):113–129, 1990.
- [13] D. Burke, S. C. Gandevia, and G. Macefield. Responses to passive movement of receptors in joint, skin and muscle of the human hand. *The Journal of physiology*, 402(1):347–361, 1988.
- [14] E. G. Jones and R. Porter. What is area 3a? *Brain Research Reviews*, 2(1):1–43, 1980.
- [15] K. J. Huffman and L. Krubitzer. Area 3a: topographic organization and cortical connections in marmoset monkeys. *Cerebral Cortex*, 11(9):849–867, 2001.

-
- [16] S. S. Kim, M. Gomez-Ramirez, P. H. Thakur, and S. S. Hsiao. Multimodal interactions between proprioceptive and cutaneous signals in primary somatosensory cortex. *Neuron*, 86(2):555–566, 2015.
- [17] J. G. Betts, P. DeSaix, E. Johnson, and J. E. Johnson. *Anatomy and Physiology*. OpenStax College, 2013.
- [18] D. J. Felleman and D. C. Van Essen. Distributed hierarchical processing in the primate cerebral cortex. *Cerebral cortex*, 1(1):1–47, 1991.
- [19] G. Bosco, R. E. Poppele, and J. Eian. Reference frames for spinal proprioception: limb endpoint based or joint-level based? *Journal of Neurophysiology*, 83(5):2931–2945, 2000.
- [20] A. Parmiggiani, M. Maggiali, L. Natale, F. Nori, A. Schmitz, N. Tsagarakis, J. S. Victor, F. Becchi, G. Sandini, and G. Metta. The design of the icub humanoid robot. *International journal of humanoid robotics*, 9(04), 2012.
- [21] T. Kohonen. Self-organized formation of topologically correct feature maps. *Biological cybernetics*, 43(1):59–69, 1982.
- [22] T. Kohonen. The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–1480, 1990.
- [23] V. Zatsiorsky and B. Prilutsky. *Biomechanics of skeletal muscles*. Human Kinetics, 2012.
- [24] Y. Kuniyoshi. 9 learning from examples: Imitation learning and emerging cognition. *Humanoid Robotics and Neuroscience: Science, Engineering and Society*, page 233, 2014.
- [25] M. D. McDonnell and N. G Stocks. Maximally informative stimuli and tuning curves for sigmoidal rate-coding neurons and populations. *Physical Review Letters*, 101(5):058103, 2008.
- [26] Y. Kuniyoshi and S. Sangawa. Early motor development from partially ordered neural-body dynamics: experiments with a cortico-spinal-musculo-skeletal model. *Biological cybernetics*, 95(6):589–605, 2006.

Príloha A

Zadanie

České vysoké učení technické v Praze
Fakulta elektrotechnická

Katedra kybernetiky

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: Naďa Bednářová

Studijní program: Otevřená informatika (bakalářský)

Obor: Informatika a počítačové vědy

Název tématu: Reprezentace proprioceptivních vstupů humanoidního robota iCub pomocí samoorganizujících se map

Pokyny pro vypracování:

1. Generování a předzpracování proprioceptivních vstupů (úhly kloubů) robota iCub či jeho simulátoru.
2. Trénování a testování různých variant algoritmu samoorganizujících se map.
3. Studium relevantních oblastí neuroanatomie a neurofyzologie týkající se primární somatosensorické kůry.
4. Porovnání reprezentace vzešlé z umělých dat s poznatky z neurovědy a adaptace algoritmu.
5. Interpretace výsledků.

Seznam odborné literatury:

- [1] Kohonen, T. (1990). The self-organizing map. *Proceedings of the IEEE*, 78(9), 1464-1480.
- [2] Krubitzer, L.; Huffman, K. J.; Disbrow, E. & Recanzone, G. (2004), "Organization of area 3a in macaque monkeys: contributions to the cortical phenotype", *Journal of Comparative Neurology* 471(1), 97--111.
- [3] Proske, U., & Gandevia, S. C. (2012). The proprioceptive senses: their roles in signaling body shape, body position and movement, and muscle force. *Physiological Reviews*, 92(4), 1651-1697.
- [4] Stafford, Tom, and Stuart P. Wilson. "Self-organisation can generate the discontinuities in the somatosensory map." *Neurocomputing* 70.10 (2007): 1932-1937.
- [5] Straka, Z. (2014): Tvorba senzorických map z taktilních vstupů robota iCub pomocí metod strojového učení. Bc. thesis, Czech Technical University, Prague, Czech Republic.

Vedoucí bakalářské práce: Mgr. Matěj Hoffmann, Ph.D.

Platnost zadání: do konce letního semestru 2014/2015

L.S.

doc. Dr. Ing. Jan Kybic
vedoucí katedry

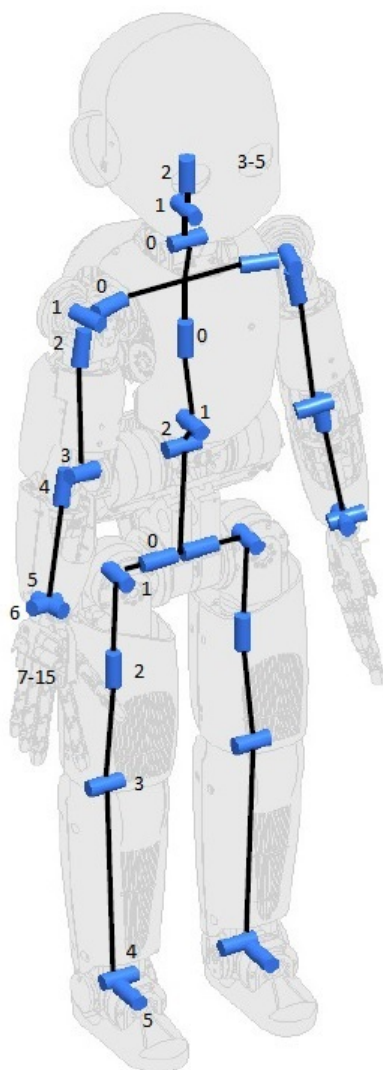
prof. Ing. Pavel Ripka, CSc.
děkan

V Praze dne 29. 1. 2015

Príloha B

Konštrukcia robota iCub

Ako som uviedla v Kapitole 3, v tabuľke B.1 uvádzam prehľad všetkých 53 stupňov voľnosti robota iCub s popisom príslušného pohybu v angličtine a s očíslovaním tak, ako sú uvedené v oficiálnej dokumentácii ¹⁾ a ako s nimi pracuje iCub simulátor ²⁾. Pre ilustráciu je väčšina SV očíslovaná v obrázku B.1



Obrázok B.1. Stupne voľnosti robota iCub

¹⁾ http://wiki.icub.org/wiki/Main_Page

²⁾ http://wiki.icub.org/wiki/ICub_Simulator_Installation

HEAD		FROM	TO	RANGE
0	neck pitch	-40	30	70
1	neck roll	-70	60	130
2	neck yaw	-55	55	110
3	eyes tilt	-35	15	50
4	eyes version	-50	52	102
5	eyes vergence	0	90	90
TORSO		FROM	TO	RANGE
0	torso yaw	-50	50	100
1	torso roll	-30	30	60
2	torso pitch	-10	70	80
LEFT(RIGHT) ARM		FROM	TO	RANGE
0	shoulder pitch	-95	10	105
1	shoulder roll	0	160,8	160,8
2	shoulder yaw	-37	80	117
3	elbow	15,5	106	121,5
4	wrist pronosupination	-90	90	180
5	wrist pitch	-90	0	90
6	wrist yaw	-20	40	60
7	hand finger adduction/abduction	0	60	60
8	thumb opposition	10	90	80
9	thumb proximal flexion/extension	0	90	90
10	thumb distal flexion	0	180	180
11	index proximal flexion/extension	0	90	90
12	index distal flexion	0	180	180
13	middle proximal flexion/extension	0	90	90
14	middle distal flexion	0	180	180
15	ring and little finger flexion	0	270	270
LEFT(RIGHT) LEG		FROM	TO	RANGE
0	hip pitch	-30	90	120
1	hip roll	0	90	90
2	hip yaw	-80	78	158
3	knee	-125	15	140
4	ankle pitch	-20	44	64
5	ankle roll	-22	22	44

Tabuľka B.1. Jednotlivé stupne kĺbov robota iCub v stupňoch.

Príloha C

Obsah CD

Použitý software:

- **Matlab** – R2014b ¹⁾
- **SOM toolbox** – ²⁾
- **iCub Simulator** – iCub_1.1.15_v10 ³⁾
- **T_EX** – L^AT_EX a plainT_EX⁴⁾

Priložené CD obsahuje nasledujúce adresáre a súbory:

- **dolianad.pdf** – text bakalárskej práce
- **doc** – priečinok so zdrojovým kódom textu v T_EX
 - **specification** – obsahuje zadanie práce
 - **pics** – obsahuje všetky obrázky použité v práci
 - **text** – obsahuje samotný text práce rozdelený do kapitol
- **source** – priečinok so zdrojovými kódmi
 - **somtoolbox** – obsahuje pôvodnú verziu SOM toolboxu
 - **data** – obsahuje použité dáta
 - **code** – obsahuje samotnú implementáciu
 - **test** – tento priečinok obsahuje scripty jednotlivých testov a ich zoznam s popisom v súbore **testy.xls**

¹⁾ <http://www.mathworks.com/products/matlab>

²⁾ <http://www.cis.hut.fi/somtoolbox/>

³⁾ http://wiki.icub.org/wiki/ICub_Simulator_Installation

⁴⁾ <https://www.tug.org/texlive/>