

bakalářská práce

Tvorba senzorických map z taktilních vstupů robota iCub pomocí metod strojového učení

Zdeněk Straka



Květen 2014

Vedoucí práce: Mgr. Michal Vavrečka, Ph.D.
Externí konzultant: Mgr. Matěj Hoffmann, Ph.D.

České vysoké učení technické v Praze
Fakulta elektrotechnická, Katedra kybernetiky

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: Zdeněk S t r a k a

Studijní program: Kybernetika a robotika (bakalářský)

Obor: Robotika

Název tématu: Tvorba sensorických map z taktilních vstupů robota iCub pomocí metod strojového učení

Pokyny pro vypracování:

1. Tvorba a testování algoritmu pro zpracování taktilních vstupů.
2. Návrh systému pro topografickou reprezentaci tělesných částí.
3. Tvorba neuronových sítí pro reprezentaci vstupů.
4. Trénování a testování variant neuronových sítí.
5. Interpretace výsledků.

Seznam odborné literatury:

- [1] Parpia, Pasha: Reappraisal of the somatosensory homunculus and its discontinuities. In: Neural Computation 23.12 (2011): 3001-3015.
- [2] John C. Pearson, Leif H. Finkel, and Gerald M. Edelman: Plasticity in the organization of adult cerebral cortical maps: a computer simulation based on neuronal group selection. In: The Journal of Neuroscience 7.12 (1987): 4209-4223.
- [3] Tom Stafford, and Stuart P. Wilson: Self-organisation can generate the discontinuities in the somatosensory map. In: Neurocomputing 70.10 (2007): 1932-1937.

Vedoucí bakalářské práce: Mgr. Michal Vavrečka, 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 10. 1. 2014

Poděkování

Rád bych upřímně poděkoval externímu konzultantovi práce Mgr. Matěji Hoffmannovi, Ph.D a vedoucímu práce Mgr. Michalu Vavrečkovi, Ph.D za jejich odbornou pomoc, vstřícnost a podporu. Dále bych chtěl poděkovat své rodině a blízkým přátelům za podporu během celého studia.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně, a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne

.....

Abstrakt

Cílem této práce je vytvořit biologicky inspirovanou reprezentaci taktilních senzorů pomocí umělé neuronové sítě. Vytvořenou sensorickou mapu by mělo být možné použít pro reprezentaci taktilních senzorů humanoidního robota iCub. Neuronová síť musí ve výstupní vrstvě odrážet topologii taktilních senzorů. Další požadovanou vlastností neuronové sítě je schopnost učit se z trénovací množiny s více současnými doteky.

Navržená architektura neuronové sítě vznikla modifikací architektury samoorganizující se mapy (SOM). Vytvořený model neuronové sítě byl otestován na trénovací množině s více současnými doteky. Tato trénovací množina byla vygenerována počítačem. Neuronová síť byla dále otestována na trénovací množině vytvořené stimulací kůže robota iCub. V tomto případě byla kůže v každém okamžiku stimulována pouze na jednom místě. V obou případech výstupní vrstva naučené neuronové sítě dobře odrážela topologii taktilních senzorů.

Dosažené výsledky jsou dobrým předpokladem pro budoucí použití této neuronové sítě k vnitřní reprezentaci kůže humanoidních robotů či k počítačovým simulacím některých neurobiologických procesů.

Klíčová slova

neuronová síť; SOM; sensorická mapa; iCub; reprezentace taktilních senzorů, umělá kůže

Abstract

The goal of this thesis is to create a bio-inspired representation of tactile sensors using an artificial neural network. The sensory map created should be used to represent tactile sensors of the humanoid robot iCub. An output layer of the neural network must reflect a topology of its artificial skin. Other desired property of the neural network is an ability to learn from a training set with more concurrent touches.

The proposed architecture of the neural network was created by modifying an architecture of self-organizing maps (SOM). The model was tested on a training set with more concurrent touches. This training set was generated by a computer. The neural network was further tested on a training set created by stimulating the skin of the iCub robot. In this case the skin was stimulated at any moment only in a one place. In both cases the output layer of the trained neural network reflects the topology of tactile sensors well.

The results achieved are a good basis for future use of the neural network for internal representation of the skin of humanoid robots or for computational simulations of some neurobiological processes.

Keywords

neural network; SOM; sensory map; iCub; representation of tactile sensors, artificial skin

Obsah

1. Úvod	1
2. Popis řešené úlohy	3
2.1. Taktilní vstupy	3
2.1.1. Biologické receptory doteku	3
2.1.2. Receptory doteku robota iCub	3
2.1.3. Repräsentace taktilních vstupů	4
2.2. Cesta signálu z kůže do sensorické mapy	4
2.3. Požadované vlastnosti reprezentace	5
2.3.1. Zachování topologie	5
2.3.2. Vliv četnosti aktivace taxelu na jeho zastoupení v mapě	5
2.3.3. Více současných doteků	5
2.3.4. Nastavení oblastí působení neuronů	6
3. Samoorganizující se mapy a jejich nedostatky pro řešenou úlohu	7
3.1. Popis architektury SOM	7
3.2. Algoritmus učení SOM	7
3.2.1. Inicializace	8
3.2.2. Soupeření	8
3.2.3. Kooperace	8
3.2.4. Adaptace	9
3.3. Nedostatky SOM pro použití v řešené úloze	10
3.3.1. Nefunkčnost použití normy rozdílu vektorů	10
Intuitivní popis problému	10
3.3.2. Nemožnost nastavení receptivních polí neuronů	11
3.3.3. Neschopnost učení se z trénovacích dat s multidoteky	11
3.4. Hraníční efekt	12
4. Modifikovaná SOM	13
4.1. Určování vítěze mSOM	13
Ilustrativní příklad	13
4.2. Souvislost mezi určováním vítěze SOM a mSOM	13
4.3. Receptivní pole	14
4.4. Normování váhového vektoru	15
4.5. Algoritmus učení mSOM	16
4.6. Určování více vítězů s využitím laterální inhibice	17
Ilustrativní příklad	17
4.6.1. Počet utlumení	17
Ilustrativní příklad - pokračování	17
4.6.2. Nastavení hodnot tlumení	18
4.6.3. Shrnutí algoritmu určování více vítězů s laterální inhibicí	18
5. Hodnocení naučených map	19
5.1. Grafické znázornění	19
5.1.1. Grafické znázornění mapy	19
Ilustrativní příklad	19
Číslování neuronů v mapě	20
Obarvení neuronů v mapě	20

	Zhodnocení použitého přiřazení	20
	Obarvení neuronů v mapě při velké kůži	21
5.1.2.	Grafické znázornění vektoru vah neuronu	22
5.2.	Hodnocení pálení neuronů v mapě	22
5.2.1.	Jeden současný dotek	23
5.2.2.	Více současných doteků	23
5.3.	Hodnocení zachování topologie	23
5.3.1.	Grafická metoda	23
5.3.2.	Číselná míra	23
	Požadavky na testovací množinu	24
	Algoritmus výpočtu	24
	Interpretace a omezení míry	24
	Možnost nastavení prahu pálení	25
5.4.	Hodnocení vlivu četnosti aktivace taxelu na jeho zastoupení v mapě	25
5.5.	Způsob hodnocení map v této práci	25
6.	Testování naučených neuronových sítí a interpretace výsledků	26
6.1.	Parametry simulací	26
6.1.1.	Parametry umělých doteků	26
	Parametr dotykPst	27
6.1.2.	Parametry map	27
	Parametry společné všem mapám	27
	Proměnné parametry	28
6.2.	Učení s umělými doteky	28
6.2.1.	Přehled sérií simulací	29
6.2.2.	1. série simulací: pálení neuronů na doteky	29
	1. simulace	29
	Interpretace a zhodnocení výsledků této série	30
6.2.3.	2. série simulací: poměr velikostí mapy a kůže	32
	1. simulace	32
	2. simulace	32
	3. simulace	34
	Interpretace a zhodnocení výsledků této série	35
6.2.4.	3. série simulací: dva současné doteky	36
	1. simulace	36
	2. simulace	37
	3. simulace	38
	4. simulace	39
	Interpretace a zhodnocení výsledků této série	40
6.2.5.	4. série simulací: vliv četnosti stimulace taxelu	40
	1. simulace	40
	Interpretace a zhodnocení výsledků této série	41
6.2.6.	5. série simulací: šest současných doteků	42
	1. simulace	42
	2. simulace	42
	3. simulace	44
	Interpretace a zhodnocení výsledků této série	44
6.2.7.	6. série simulací: nastavení receptivních polí, velká kůže	45
	1. simulace	45
	2. simulace	45

3. simulace	47
Interpretace a zhodnocení výsledků této série	47
6.3. Učení s doteky z robota iCub	48
6.3.1. Popis trénovacích dat	48
6.3.2. Přehled simulací	49
6.3.3. 1. simulace: dlaň	49
Interpretace a zhodnocení výsledků této simulace	51
6.3.4. 2. simulace: dlaň a předloktí	51
Interpretace a zhodnocení výsledků této simulace	52
6.3.5. 3. simulace: přední část trupu	52
Interpretace a zhodnocení této simulace	54
6.3.6. 4. simulace: homunculus	55
Parametry simulace	55
Znázornění	55
Interpretace a zhodnocení této simulace	55
7. Závěr	57
Přílohy	
A. Algoritmus předzpracování taktilních vstupů	59
A.1. Formát vstupních dat z iCuba	59
A.2. Funkce předzpracování taktilních vstupů	60
A.2.1. Funkce data2actTax	60
Příklad použití	60
A.2.2. Funkce loadActTaxels	61
Offset částí kůže	61
Příklad použití	61
B. mSOM v Matlabu	63
B.1. Funkce pro učení mSOM	63
B.2. Funkce pro určení nejsilněji pálicího neuronu	64
B.3. Funkce pro určení nejsilněji pálicích neuronů s laterální inhibicí	64
C. Kůže robota iCub	65
C.1. Části kůže robota iCub	65
C.1.1. Dlaň	65
C.1.2. Předloktí	65
C.1.3. Nadloktí	66
C.1.4. Přední část trupu	66
C.2. Návaznost částí kůže	66
D. Obsah přiloženého CD	69
Literatura	70

1. Úvod

Cílem této práce je vytvořit biologicky inspirovanou reprezentaci taktilních senzorů pomocí umělé neuronové sítě. Vytvořený způsob reprezentace taktilních senzorů by měl být použitelný pro humanoidního robota iCub [1]. Trénovací data vytvořená na robotovi iCub jsem získal od externího konzultanta této práce Matěje Hoffmanna.



Obrázek 1. Robot iCub (foto Laura Taverna)

Nebudu se snažit přesně modelovat biologické sensorické mapy (definice pojmu viz 2.2). Vytvořit přesný biologický model sensorické mapy pro reprezentaci taktilních senzorů by bylo velmi náročné a s dostupnými výpočetními prostředky pravděpodobně nerealizovatelné. Dalším omezením v přesnosti modelu jsou samotné taktilní vstupy vytvořené buď uměle, nebo z robota iCub, které pochopitelně nejsou zcela ekvivalentní biologickým taktilním vstupům. V neposlední řadě současná neurověda zatím není schopna podat zcela přesný popis fungování těchto neurobiologických struktur.

Přes všechny uvedené překážky je možné vytvořit umělou sensorickou mapu pro reprezentaci taktilních senzorů, která funguje na stejných základních principech (alespoň na některých) jako biologická sensorická mapa. Takový přístup umožňuje lépe pochopit tyto základní principy. Pochopení těchto principů pak může vést k lepšímu pochopení samotných biologických sensorických map. Přínos takového přístupu může být značný, protože umožňuje provádět experimenty, které buď nelze s živými bytostmi vůbec provádět, nebo to je například z etických důvodů problematické.

Kromě snahy o lepší porozumění biologickým sensorickým mapám by tato práce měla být také přínosná v experimentální robotice. Podle přístupu "pochopení skrze tvoření"[2] se budu snažit, aby navržená neuronová síť mohla sloužit k reprezentaci taktilních senzorů reálného robota. Biologicky inspirovaný přístup umožňuje vytvořit vnitřní reprezentaci robotické kůže, která dobře odráží strukturu kůže, pouze stimulací kůže vhodnými doteky. To znamená, že není potřeba k vytvoření takové reprezentace

1. Úvod

žádné měření nebo identifikování jednotlivých senzorů uvnitř kůže.

Řešená úloha spočívá v návrhu architektury umělé neuronové sítě. Neuronová síť se natrénuje z trénovací množiny taktilních vstupů. Takto vzniklá reprezentace kůže by měla co nejvíce splňovat vlastnosti, které uvedu dále v textu.

V kapitole 2 podrobněji popíši řešenou úlohu včetně souvislostí s její biologickou inspirací. V kapitole 3 se budu zabývat samoorganizujícími se mapami, ze kterých vychází navržená architektura neuronové sítě pro reprezentaci taktilních senzorů, která je tématem kapitoly 4. Způsobem hodnocení naučených sensorických map se budu zabývat v kapitole 5. Naučené mapy budu prezentovat a interpretovat v kapitole 6. V kapitole 7 provedu zhodnocení dosažených výsledků této práce.

Algoritmy pro předzpracování dat z robota iCub jsou uvedeny v příloze A.

2. Popis řešené úlohy

V této kapitole popíši podrobněji řešenou úlohu. Budu se snažit ukázat souvislost řešené úlohy s její biologickou inspirací. Dále uvedu vlastnosti, které by měla vytvořená neuronová síť co nejvíce splňovat.

Pro snadnější vyjadřování budu dále používat slovo kůže ve smyslu taktilních senzorů v kůži. Napíši-li dále v textu například *reprezentace kůže*, je tím myšlena reprezentace senzorů v kůži citlivých na dotek, nikoliv reprezentace kůže jako orgánu pokrývající lidské tělo.

2.1. Taktilní vstupy

V této podkapitole popíši nejdříve biologické receptory doteku. Následně uvedu jejich robotický ekvivalent a způsob reprezentace taktilních vstupů použitý v této práci.

2.1.1. Biologické receptory doteku

Schopnost diskriminativního doteku (*discriminative touch*) je jedním ze čtyř hlavních způsobů somatického vnímání (*somatic sensibility*). Je realizována typem neuronů nazývaných dorsální kořenová ganglia (*dorsal root ganglion neurons*) [3]. Tyto neurony zajišťují dvě základní funkce:

- Transdukcii stimulů,
- přenos zakódovaného stimulu do centrálního nervového systému.

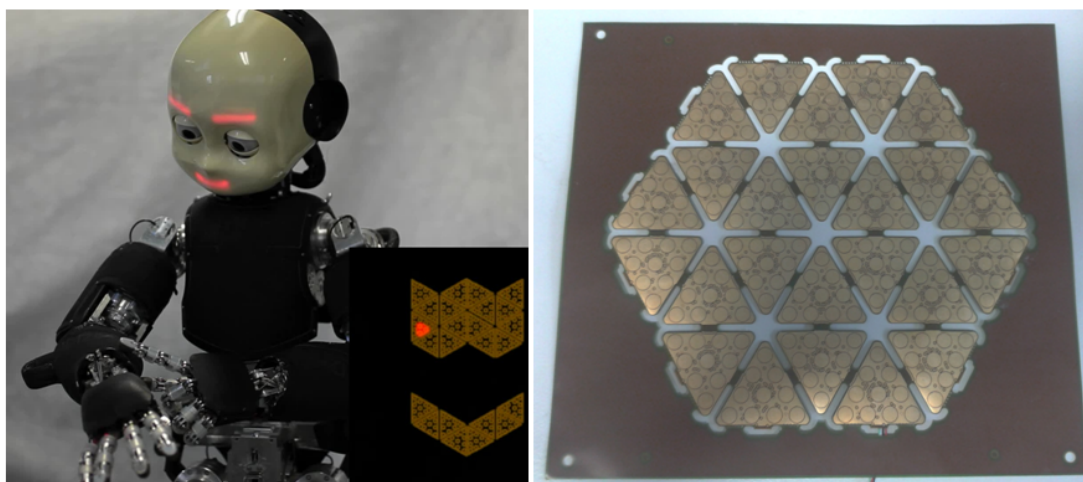
Pro vnímání podnětů z kůže existuje více receptorů. V této práci se zaměřím pouze na vnímání doteků ve smyslu tlakového působení na povrch kůže. To zajišťují receptory nazývané Merkelovy disky. Receptivní pole jednoho Merkelova disku má přibližně kruhový tvar s průměrem od 2 do 10 milimetrů. Jeden dorsální kořenový ganglion přijímá na vstupu signály z 10 až 25 Merkelových disků [3].

2.1.2. Receptory doteku robota iCub

Zde uvedu způsob realizace receptorů doteku u robota iCub [4]. Počítačová reprezentace kůže pro vytváření umělých trénovacích dat bude mít obdobný charakter.

Robot iCub je pokryt na dlaních, předloktích, nadloktích a přední části trupu umělou kůží. Kůže se skládá z několika plátů (*patches*). Z plátů jsou tvořeny jednotlivé části kůže (např. předloktí je tvořeno dvěma pláty). Části kůže na sebe ne vždy navazují. Například mezi kůží předloktí a nadloktí je mezera nepokrytá kůží (kloub). Více informací o geometrii plátů a částí kůže je uvedeno v příloze C.

Uvedené pláty kůže se skládají z malých senzorů citlivých na dotek (tlakové působení). Tyto malé senzory, základní elementy robotické kůže, budu dále nazývat taxely. Snímací frekvence taxelů je 50 hertzů. Taxely jsou uspořádány uvnitř trojúhelníkových částí. Jeden trojúhelník obsahuje celkem 10 taxelů. Celkový počet taxelů na kůži iCuba je přibližně 1750. Průměr jednoho taxelu je 4 milimetry.



Obrázek 2. Na obrázku vlevo se iCub dotýká svého předloktí. Dole vpravo na stejném obrázku lze vidět, které taxely jsou stimulovány. Na obrázku vpravo je detail ohebné desky s taxely (senzory doteku), ze které se tvoří kůže. Deska se skládá z částí trojúhelníkového tvaru. Každý trojúhelník obsahuje 10 taxelů.

Je vidět, že velikost taxelu je řádově stejná jako velikost receptivního pole jednoho Merkelova disku. Dále taxely reagují na tlakové působení obdobně jako Markelovy disky. V uvedených ohledech se dá na taxel dívat jako na robotickou obdobu Merkelova disku.

2.1.3. Reprezentace taktilních vstupů

V této práci rozlišuji pouze dva možné stavy taxelu. Jeden stav odpovídá stimulaci daného taxelu. Druhý stav odpovídá situaci, kdy taxel není stimulován. Intenzita tlakového působení se nebude rozlišovat. Stav stimulace taxelu budu reprezentovat hodnotou 1, druhý stav hodnotou 0. Stav všech taxelů v daném okamžiku budu reprezentovat vektorem, ve kterém jednotlivé složky představují stavy jednotlivých taxelů. Jinak řečeno, tyto vektory taktilních vstupů reprezentují přesně místa doteků na robotické kůži.

2.2. Cesta signálu z kůže do sensorické mapy

Dorsální kořenová ganglia nejdříve přijímají signál z Merkelových disků. Následně nervová vlákna z dorsálních kořenových ganglií vedou ve svazcích do míchy. Pak vedou částí míchy medial lemniscus do thalamu. Z thalamu pokračují do somatosenzorické kůry.

Oblast mozku, která reaguje na smyslovou stimulaci a její prostorová organizace odráží některé vlastnosti smyslové stimulace, se nazývá sensorická mapa. Sensorická mapa pro reprezentaci kůže je její topografickou reprezentací. Nachází se v Brodmannově oblasti 3b primární somatosenzorické kůry [3].

V této úloze, pro zjednodušení, vedou taxely přímo do sensorické mapy. To znamená, že neuronová síť bude mít pouze vstupní a výstupní vrstvu. Vstupní vrstvu představují vektory taktilních vstupů. Výstupní vrstva představuje sensorickou mapu.

2.3. Požadované vlastnosti reprezentace

V této podkapitole popisují vlastnosti, které by vzniklá reprezentace kůže v sensorické mapě měla co nejvíce splňovat. Splnění požadovaných vlastností závisí také na trénovací množině. Proto je nutné, aby se neuronová síť učila na vhodných trénovacích datech. Některé vlastnosti množiny trénovacích dat mohou vést ke splnění některých požadovaných vlastností a zabránit splnění ostatních požadovaných vlastností. To však může být někdy pro pochopení mechanismů učení neuronové sítě užitečné.

2.3.1. Zachování topologie

V Brodmannově oblasti 3b primární somatosenzorické kůry je reprezentována kůže z celého povrchu těla. Tato sensorická mapa se nazývá homunculus. Neurony této oblasti páli podle toho, jaké části kůže jsou stimulovány a jakým způsobem jsou stimulovány. Uspořádání neuronů je až na pár diskontinuit takové, že vzájemně blízké oblasti kůže jsou reprezentovány neurony, které jsou také blízko sebe. To znamená, že se v reprezentaci kůže v sensorické mapě zachovává topologie kůže. Někdy pro toto zachování topologie kůže v sensorické mapě budu používat označení zachování topologie vstupní vrstvy ve výstupní vrstvě nebo jen krátce zachování topologie.

Jak jsem již zmínil, existují v sensorické mapě i dvě významné diskontinuity. Konkrétně ruce (prsty, dlaně a záprstí) jsou reprezentovány vedle obličeje. Chodidla jsou reprezentována vedle genitálií. Existuje hypotéza, že tento jev je způsoben častými koaktivacemi kůže těchto oblastí během vývoje plodu [5]. Časté koaktivace v trénovacích datech mohou vést u samoorganizujících se map ke vzniku diskontinuit [6]. V této práci se nebudu pokoušet tento jev modelovat. Proto se budu snažit častým koaktivacím v trénovacích datech vyhnout.

2.3.2. Vliv četnosti aktivace taxelu na jeho zastoupení v mapě

V Brodmannově oblasti 3b lze pozorovat, že počet neuronů reprezentující jednotlivé oblasti kůže není úměrný ploše těchto oblastí kůže. Počet reprezentujících neuronů pro jednotlivé oblasti kůže závisí hlavně na hustotě receptorů v těchto oblastech. Hustota receptorů je větší u částí kůže, které hrají větší význam pro smyslové vnímání. Pro představu centimetr čtvereční kůže na prstech zabírá v mapě stokrát větší oblast než centimetr čtvereční kůže na břichu [3].

U robota iCub je hustota receptorů (taxelů) všech částí kůže stejná. Předpokládejme, že části kůže významnější pro smyslové vnímání, jsou oproti ostatním částem kůže více stimulované. Stejně jako v biologickém případě je žádoucí, aby významnější části kůže byly reprezentované větším počtem neuronů než méně významné části kůže. Proto by v uměle vytvořené sensorické mapě měly být více reprezentovány části kůže, které jsou častěji stimulované. Tato vlastnost závisí na způsobu sběru dat. Ve své práci se této vlastnosti nebudu moc věnovat. Provedu pouze jednu simulaci pro ilustraci toho, že navržená neuronová síť tuto vlastnost splňuje. Pro zkoumání a vyhodnocování ostatních vlastností je spíše výhodné, když jsou oblasti kůže stimulovány rovnoměrně.

2.3.3. Více současných doteků

U člověka je běžné, že je v jednom okamžiku stimulováno více od sebe vzdálených oblastí kůže. Uměle vytvořená sensorická mapa by proto měla být schopná se učit z vektorů taktálních vstupů, které obsahují více současných stimulací na vzájemně vzdálených oblastech kůže.

2. Popis řešené úlohy

Ve fázi po naučení by v případě vstupního vektoru s více současnými doteky mělo být v sensorické mapě více vítězných neuronů. Vítězné neurony by měly reprezentovat stimulované části kůže v daný okamžik. V případě více současných doteků bude pro lepší odlišení neuronů, které reprezentují různé stimulované části kůže, výhodné použít princip laterální inhibice.

2.3.4. Nastavení oblastí působení neuronů

Dalším požadavkem je možnost pro jednotlivé části sensorické mapy nastavit oblasti kůže, na které mají během učení reagovat. To znamená, že je možné určitě skupině neuronů nastavit to, že bude během učení ovlivňována pouze doteky z určité oblasti kůže (např. z předloktí). Umožní to provést určité přednastavení sensorické mapy, které pomůže učení. Protože se nesnažím o simulování učení a tvorby sensorické mapy během vývoje plodu či dokonce evolučního vývoje, může být určité přednastavení sensorické mapy na místě.

3. Samoorganizující se mapy a jejich nedostatky pro řešenou úlohu

Protože výsledná neuronová síť pro reprezentaci taktilních sensorů vznikla modifikací samoorganizujících se map, popíšu v této kapitole tento druh neuronové sítě.

Samoorganizující se mapy (SOM) jsou druhem umělých neuronových sítí. Tento druh neuronové sítě se často označuje po svém autorovi Teuvo Kohonenovi [7] jako Kohonenovy sítě nebo Kohonenovy mapy. Pro tyto neuronové sítě je charakteristické učení bez učitele. To znamená, že SOM během učení nemá žádné informace o tom, jaké neurony mají na dané vstupy reagovat.

Tento druh neuronové sítě transformuje spojitý vstupní prostor vysoké dimenze do diskrétního výstupního prostoru nižší dimenze (často dvoudimenzionálního). Velmi důležitou schopností SOM je, že se při splnění určitých podmínek vzájemně blízké vstupní vektory (ve smyslu Euklidovské normy jejich rozdílů) zobrazují na vzájemně blízké neurony ve výstupní vrstvě SOM. Jinak řečeno, ve výstupní vrstvě se zachovává topologie vstupní vrstvy. SOM jsou díky svým vlastnostem vhodné pro modelování některých neurobiologických struktur [8].

3.1. Popis architektury SOM

V následujícím textu budu pojmem mapa označovat výstupní vrstvu neuronové sítě. Dále budu uvažovat dvoudimenzionální mapu. To znamená, že výstupní vrstva mapy má neurony uspořádané v rovině.

Na vstupu mapy je vstupní vektor

$$\mathbf{x} = [x_1, \dots, x_n]^T \in \mathbb{R}^N.$$

Ve výstupní vrstvě mapy je m neuronů uspořádáno v mřížce a každému neuronu je přiřazen váhový vektor

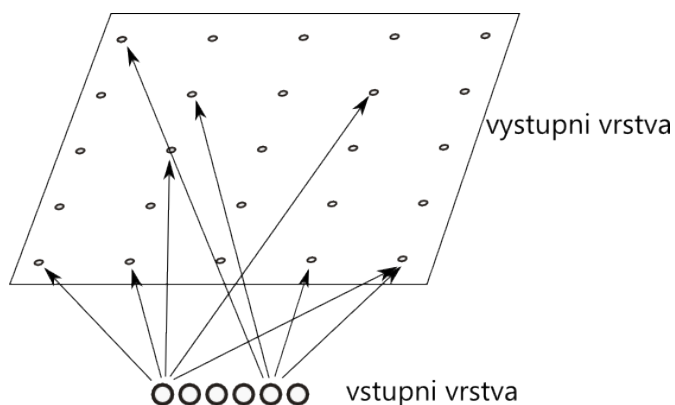
$$\mathbf{w}_i = [w_{i,1}, \dots, w_{i,n}]^T \in \mathbb{R}^N, \quad i = 1, 2, \dots, m.$$

Architektura SOM je schématicky znázorněna na obr. 3. Na tomto obrázku není z důvodu přehlednosti znázorněno to, že každý neuron v mapě je propojen se všemi složkami vektoru vstupní vrstvy. Každému propojení mezi složkou vstupního vektoru a neuronem lze podle příslušného vektoru vah přiřadit určitou váhu.

3.2. Algoritmus učení SOM

Učení lze rozdělit na čtyři části, které můžeme nazvat jako inicializace, soupeření, kooperace, adaptace. Inicializace se provádí pouze na začátku. Části soupeření, spolupráce a adaptace se cyklicky provádí pro každý vybraný vstupní vektor z trénovací množiny. Ukončení může být určeno třeba počtem výběrů vstupních vektorů z trénovací množiny. V tomto případě by měl být počet takových výběrů minimálně desetkrát větší než počet neuronů mapy [9]. Další možností je ukončit učení, když se budou měnit váhové vektory pouze zanedbatelně [10].

3. Samoorganizující se mapy a jejich nedostatky pro řešenou úlohu



Obrázek 3. Schématické znázornění vrstev SOM

3.2.1. Inicializace

Možností jak inicializovat váhové vektory jednotlivých neuronů je více. Základní možností je náhodná inicializace. V tomto případě jsou za jednotlivé složky váhových vektorů dosazena například náhodně generovaná čísla v rozsahu hodnot složek vstupních vektorů [10].

3.2.2. Soupeření

Nejdříve se náhodně vybere vstupní vektor \mathbf{x} z trénovací množiny. Pak se určí neuron $j \in \{1, 2, \dots, m\}$, jehož váhový vektor \mathbf{w}_j je, obrazně řečeno, nejvíce podobný vstupnímu vektoru. Vítězný neuron j se určí

$$j(\mathbf{x}) = \arg \min_i \|\mathbf{w}_i - \mathbf{x}\|_2^2, \quad i = 1, 2, \dots, m.$$

Z toho lze usoudit, že intuitivní představa podobnosti vstupního a váhového vektoru v matematickém vyjádření znamená malou hodnotu euklidovské normy rozdílu $\|\mathbf{w}_i - \mathbf{x}\|_2^2$.

3.2.3. Kooperace

Kooperace vychází z neurobiologického pozorování laterální interakce mezi vybuzenými neurony [10]. Neuron, který pálí, stimuluje také neurony ve svém okolí. Čím dále jsou neurony od tohoto pálicího neuronu, tím je tato stimulace slabší. Tohoto principu se využívá i u SOM. Proto se zavádí funkce okolí $h : \{1, 2, \dots, m\} \times \{1, 2, \dots, m\} \rightarrow \mathbb{R}$. Funkce $h(i, j)$ určuje rozsah kooperace mezi vítězným neuronem j a dalšími neurony $i = 1, 2, \dots, m$. Dále se zavádí funkce $d : \{1, 2, \dots, m\} \times \{1, 2, \dots, m\} \rightarrow \mathbb{R}$. Funkce $d(i, j)$ vyjadřuje vzdálenost neuronů i a j v mřížce. Jako $d(i, j)$ se často bere euklidovská vzdálenost souřadnic neuronů i a j v mřížce. Dále se předpokládá $h(i, j(\mathbf{x}))$ jako funkce $d(i, j)$.

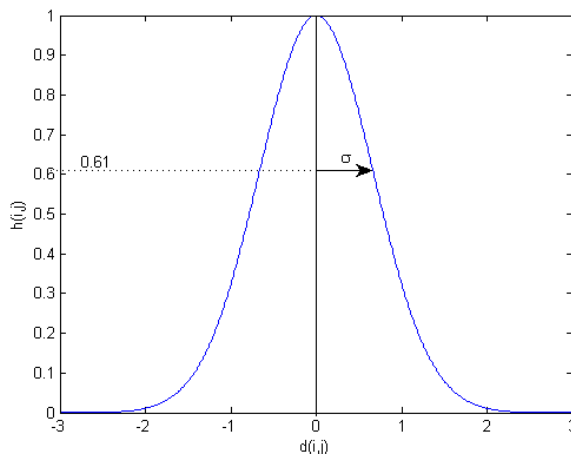
Funkce okolí musí pak splňovat tyto vlastnosti [10]:

- Funkce $h(i, j(\mathbf{x}))$ musí být symetrická kolem vítězného neuronu $j(\mathbf{x})$.
- Funkce $h(i, j(\mathbf{x}))$ monotonně klesá s rostoucí vzdáleností $d(i, j)$. Pro $d(i, j) \rightarrow \infty$ jde $h(i, j(\mathbf{x}))$ k 0.

Tyto vlastnosti splňuje Gaussova funkce

$$h(i, j(\mathbf{x}), n) = \exp\left(-\frac{d^2(i, j(\mathbf{x}))}{2\sigma^2(n)}\right),$$

kde $\sigma(n)$ je parametr, který klesá s časem k nule a tím zajišťuje zmenšování vlivu vítězného neuronu na okolní neurony během učení. Parametr $n \in \mathbb{N}$ představuje n -tý výběr vstupního vektoru z trénovací množiny (s časem roste). Za povšimnutí stojí, že funkce okolí $h(i, j(\mathbf{x}), n)$ je také funkcí parametru n . Bylo by možné zvolit tuto funkci jako konstantní na určitém okolí vítězného neuronu a dále jako nulovou, ale Gaussova funkce je z neurobiologického pohledu vhodnější [10]. Průběh Gaussovy funkce se znázorněným významem parametru σ je vykreslen na obr. 4.



Obrázek 4. Gaussova funkce s vyznačeným parametrem σ

3.2.4. Adaptace

Nejdříve intuitivním způsobem vysvětlím, co je zde myšleno adaptací. Následně to matematicky zformalizuji.

Vybere se vstupní vektor z trénovací množiny. Následně se určí vítězný neuron, který reaguje na vybraný vstupní vektor nejsilněji. Adaptace pak znamená, že vítězný neuron a jemu blízké neurony se stanou podobnější vybranému vstupnímu vektoru. To znamená, že pokud následně dáme na vstup ten stejný váhový vektor (či jemu podobný), budou na něj adaptované neurony reagovat o něco silněji. Matematicky to znamená, že euklidovská vzdálenost mezi adaptovanými neurony a vstupním vektorem bude o něco menší. Přesně tohle zajistí následující způsob adaptace váhových vektorů

$$\mathbf{w}_i(n+1) = \mathbf{w}_i(n) + \alpha(n)h(i, j(\mathbf{x}), n)(\mathbf{x} - \mathbf{w}_i(n)), \quad i = 1, 2, \dots, m,$$

kde parametr $\alpha(n)$ se nazývá rychlost učení a s časem klesá k nule. Rychlost učení může být například definována jako

$$\alpha(n) = \frac{A}{n+B},$$

kde A, B jsou vhodně zvolené konstanty.

Dá se říct, že díky adaptaci se během učení z váhových vektorů jednotlivých neuronů stávají prototypy vstupních vektorů z trénovací množiny. Tak vzniká reprezentace vektorů ze vstupního prostoru v prostoru výstupní vrstvy. Pro úplnost ještě uvedu, že většinou je počet neuronů menší než počet možných vstupních vektorů, a proto není v těchto případech ani teoreticky možné, aby každý vstupní vektor byl přesně reprezentován nějakým neuronem. Účelem SOM není přesně reprezentovat vstupní vektory.

3.3. Nedostatky SOM pro použití v řešené úloze

V této podkapitole uvedu důvody, proč nebylo vhodné použít klasickou SOM pro reprezentaci taktálních senzorů. Nejdříve stručně připomenu podstatné vlastnosti vstupní neuronové sítě v řešené úloze. Vstupní vektory reprezentující stimulaci kůže robota iCub mají více jak tisíc složek. Každá složka vstupního vektoru reprezentuje jeden taxel. Složka odpovídající určitému taxelu má hodnotu 1 pokud je daný taxel aktivován, jinak má hodnotu 0. Počet aktivovaných taxelů se v jednom okamžiku pohybuje od jednotek až po desítky taxelů. To znamená, že vstupní vektory jsou řídké. V jednom okamžiku může být kůže stimulována na několika od sebe vzdálených místech současně.

3.3.1. Nefunkčnost použití normy rozdílu vektorů

Použitá norma rozdílu vektorů u klasické SOM neodráží dobře počet společně koaktivovaných taxelů ve vstupních vektorech s doteky. Protože SOM zachovává topologii vstupní vrstvy ve výstupní vrstvě (při splnění určitých podmínek) ve smyslu normy rozdílu dvou vektorů, není pro řešenou úlohu vhodná. To znamená, že vzájemně podobné doteky s velkým počtem stejných aktivovaných taxelů nemusí být reprezentované na mapě blízko sebe. Pro řešenou úlohu bylo nutné navrhnout neuronovou síť, která zachovává "topologii koaktivací". To znamená, že jsou-li taxely často koaktivované, pak i doteky v nichž hrají významnou roli tyto taxely, budou reprezentované na mapě blízko sebe.

Problém je poměrně detailně popsán, intuitivním způsobem usnadňující pochopení, v následujících odstavcích.

Intuitivní popis problému

Nejdříve se podívám na typickou úlohu, která se řeší pomocí SOM. Budu měřit na nějakém objektu několik veličin a pomocí hodnot těchto veličin mám tento objekt správně zařadit do určité třídy. Pro jednoduchost mám list stromu, u něhož měřím pouze výšku, šířku a hmotnost a tyto veličiny budou nabývat řádově stejných hodnot (abych nemusel normovat). To znamená, že budu mít vektor $\mathbf{v} = (\text{výška}, \text{šířka}, \text{hmotnost})$. Vezmu-li vektory dvou listů, tak hned mám nějakou přibližnou intuitivní představu o tom jestli jsou si ty dva listy podobné nebo nikoliv. Tento dojem bude pravděpodobně docela poměrně dobře odrážet hodnota normy rozdílu těchto dvou vektorů $\|\mathbf{v}_1 - \mathbf{v}_2\|_2$. Dále intuitivně očekávám, že když tyto dva listy budou mít podobné rozměry a hmotnost, tak budou také reprezentované na mapě blízko sebe.

Teď uvažuji případ úlohy kterou řeším, kdy mám vektor doteku, ve kterém je pár jedniček a zbytek tvoří velký počet nul. Pro jednoduchost předpokládám dále jen vstupní vektory, které tvoří jeden souvislý dotek rozumné velikosti (několik taxelů). Takový vektor budu označovat jako vektor s jednodotekem. Vezmu-li si dva takové vstupní vektory, nejsem z nich schopen (v obecném případě) poznat, zda jsou tyto dva doteky na kůži blízko sebe nebo nikoliv. Pouze z těchto dvou vektorů nemám žádnou představu o tom, kde ty doteky na kůži leží (pokud neznám pozice taxelů). Jedinou informací, kterou mám o vzdálenosti těchto dvou doteků je to, kolik mají společných jedniček v jim příslušících vektorech (taxelů, které jsou aktivované v obou dotecích). Protože váhové vektory neuronů se dají vnímat jako určité prototypy vstupních vektorů, provedené úvahy se dvěma vektory ze vstupního prostoru budou v principu platné i v případě vstupního a váhového vektoru. To ilustruji na jednoduchém příkladu:

Mám vstupní vektor

$$\mathbf{x} = (1, 0, 0, 0, 0, \dots, 0)$$

a dva váhové vektory neuronů

$$\mathbf{w}_1 = (0, 0, 0, 0, 1, 0, \dots, 0), \mathbf{w}_2 = (1, 1, 1, 1, 0, 0, \dots, 0).$$

Pak

$$\|\mathbf{x} - \mathbf{w}_1\|^2 = 2, \|\mathbf{x} - \mathbf{w}_2\|^2 = 3$$

To znamená, že na základě normy rozdílu váhového a vstupního vektoru vyhraje neuron s váhovým vektorem \mathbf{w}_1 (hledám minimum). To není žádoucí výsledek, protože neuron s \mathbf{w}_1 na rozdíl od neuronu s \mathbf{w}_2 nemá s \mathbf{x} ani jeden společně aktivovaný taxel.

Teď už začíná být jasné, proč je použití normy rozdílu používané u klasické SOM v mnou řešené úloze velmi nevhodné. Norma rozdílu dvou vektorů s jednodoteky vyjadřuje, kolik je v obou dotecích aktivováno stejných taxelů velmi špatně a už vůbec nevyjadřuje, jak moc jsou od sebe doteky vzdáleny na kůži. Proto je použití normy rozdílu vstupního a váhového vektoru při určování vítěze pro řešenou úlohu nevhodné.

Klasická SOM za určitých podmínek zachovává topologii vstupní vrstvy ve výstupní vrstvě. Proto dva podobné lístky (z pohledu normy rozdílu jejich vektorů) budou blízko sebe reprezentované i na mapě. Teď si vezmu případ dvou sousedních doteků, které se nebudou vůbec překrývat (třeba stimulace jednoho taxelu a následně druhého tak, že oba taxely jsou vedle sebe). Přestože z pohledu kůže jsou doteky velmi blízko, SOM vůbec nepozná, že spolu nějak sousedí. Tedy není vůbec obecně garantováno, že budou vedle sebe i na mapě. Tak to bude v případě, že budou společně koaktivované nebo jejich sousední taxely, se kterými jsou společně koaktivované, budou společně koaktivované. Navržená neuronová síť proto musí splňovat, že jsou-li taxely často koaktivované, pak i doteky, v nichž hrají významnou roli tyto taxely, budou reprezentované na mapě blízko sebe.

3.3.2. Nemožnost nastavení receptivních polí neuronů

Mělo by být možné pro jednotlivé neurony nastavit oblasti kůže, na něž mají během učení reagovat (viz 2.3.4). Oblast kůže, na kterou neuron během učení reaguje, budu nazývat receptivní pole. Je důležité si uvědomit, že podle této definice se pod pojmem receptivní pole neskrývá celá oblast kůže, na kterou daný neuron po naučení skutečně reaguje. Neuron by měl po naučení nezanedbatelně reagovat pouze na některé taxely ve svém receptivním poli. Pokud by se velikost receptivních polí jednotlivých neuronů nastavila dostatečně malá, tak by po naučení mohl neuron skutečně nezanedbatelně reagovat na všechny taxely svého receptivního pole.

Většinou se pojmem receptivní pole označuje oblast kůže, na níž neuron po naučení skutečně nezanedbatelně reaguje. V této práci nebudu pojem receptivního pole v tomto významu používat. Obecně neplatí, že by neuron po naučení reagoval skutečně na všechny taxely, na které mohl reagovat během učení.

Vhodné nastavení receptivních polí by mělo učení usnadnit a zvýšit robustnost vůči více současným dotekům na různých částech kůže. Neurony s nastaveným receptivním polem nebudou pak ovlivňovány doteky mimo jejich receptivní pole, ale pouze doteky v jejich receptivním poli.

U klasické SOM nelze pro každý neuron zvlášť nastavit receptivní pole. Proto je pro řešenou úlohu použití klasické SOM nevhodné.

3.3.3. Neschopnost učení se z trénovacích dat s multidoteky

Dalším důvodem, proč nelze použít klasickou SOM, je neschopnost učit se ze vstupních vektorů, které obsahují více současných doteků - multidoteky (viz 2.3.3). V případě

3. Samoorganizující se mapy a jejich nedostatky pro řešenou úlohu

použití trénovacích dat s multidoteky u klasické SOM není, podle provedených simulací, ve výstupní vrstvě zachována topologie vstupní vrstvy (viz 2.3.1). Protože chci, aby se neuronová síť byla schopna učit i z vektorů obsahujících rozumný počet více současných doteků, je použití klasické SOM nevhodné.

3.4. Hraniční efekt

Poměrně důležitým jevem, se kterým se lze setkat u natrénované SOM, je hraniční efekt.

Přidáme-li váhové vektory do prostoru s vektory ze vstupního prostoru (to lze, mají stejnou dimenzi) lze pozorovat, že váhové vektory neuronů jsou mírně kontrahované směrem dovnitř [11]. Jev je způsoben tím, že krajní neurony jsou v důsledku asymetrie okolí v průměru víc adaptované dovnitř sítě.

4. Modifikovaná SOM

V této kapitole se budu zabývat popisem navrženého modelu neuronové sítě pro reprezentaci taktilních sensorů. Tento model musí splňovat požadavky uvedené v 2.3.

Navržený model neuronové sítě vznikl modifikací klasické SOM (viz 3). Proto jej dále budu označovat jako mSOM. Architektura mSOM se od klasické SOM moc neliší. Učení probíhá podobným způsobem. Podstatný rozdíl je v určování vítěze, v možnosti nastavení receptivních polí a v normování. Nejdříve popíšu každou z těchto modifikací samostatně. Následně uvedu celý algoritmus učení. Na závěr uvedu způsob určování více vítězů po naučení mSOM s využitím laterální inhibice.

Kapitolu, která se zabývá implementací mSOM v Matlabu, lze nalézt v příloze B.

4.1. Určování vítěze mSOM

V 3.3.1 jsem uvedl důvody, proč není vhodné používat při určování vítěze normu rozdílu váhového a vstupního vektoru. Základní myšlenku ve stručnosti uvedu ještě jednou. Představíme-li si váhové vektory principiálně jako prototypy doteků, tak potřebujeme určit, do jaké míry se tento prototyp doteku překrývá s dotekem ve vstupním vektoru. Jak bylo řečeno, tuto míru norma rozdílu vektoru vah a vstupního vektoru vyjadřuje špatně (beru v potaz charakter vstupních vektorů). Daleko lépe se pro tento účel hodí využít skalární součin vstupního vektoru a vektoru vah.

Ilustrativní příklad

Pro představu uvažujme vektory vah

$$\mathbf{w}_i \in \{0, 1\}^N, \quad i = 1, 2, \dots, m,$$

kde m je počet neuronů a hodnoty složek nabývají pouze binárních hodnot. Hodnota 1 bude vyjadřovat propojení neuronu s taxelem, naopak hodnota 0 bude vyjadřovat, že taxel s daným neuronem není vůbec propojený. Pak skalární součin vstupního vektoru $\mathbf{x} \in \{0, 1\}^N$ a vektoru vah \mathbf{w}_i

$$p = \mathbf{w}_i^T \mathbf{x}$$

přímo vychází jako počet složek p , které mají hodnotu 1 ve vektoru vah \mathbf{w}_i i ve vstupním vektoru \mathbf{x} .

Z tohoto příkladu je vidět, že skalární součin poměrně dobře odráží, do jaké míry se dotek ve vstupním vektoru překrývá s prototypem doteku reprezentovaným váhovým vektorem neuronu. Ve skutečnosti složky váhového vektoru nenabývají pouze binárních hodnot, ale prezentovaný princip (dobrého vyjádření míry překryvu) zůstane i tak zachován.

4.2. Souvislost mezi určováním vítěze SOM a mSOM

Nyní ukážu souvislost mezi určením vítěze pomocí skalárního součinu a určením vítěze pomocí normy rozdílu vstupního a váhového vektoru.

Pomocí normy rozdílu se vítězný neuron j určuje jako

$$j(\mathbf{x}) = \arg \min_i \|\mathbf{w}_i - \mathbf{x}\|_2^2, \quad i = 1, 2, \dots, m.$$

Ekvivalentními úpravami

$$\|\mathbf{w}_i - \mathbf{x}\|_2^2 = (\mathbf{w}_i - \mathbf{x})^T (\mathbf{w}_i - \mathbf{x}) = \mathbf{w}_i^T \mathbf{w}_i - 2\mathbf{w}_i^T \mathbf{x} + \mathbf{x}^T \mathbf{x} = \|\mathbf{w}_i\|_2^2 - 2\mathbf{w}_i^T \mathbf{x} + \|\mathbf{x}\|_2^2$$

jsem dostal rovnost

$$\|\mathbf{w}_i - \mathbf{x}\|_2^2 = \|\mathbf{w}_i\|_2^2 - 2\mathbf{w}_i^T \mathbf{x} + \|\mathbf{x}\|_2^2.$$

U poslední rovnosti si lze všimnout, že kromě vlivu skalárního součinu je určení vítěze ovlivněno ještě členem $\|\mathbf{w}_i\|_2^2$. Protože hledám minimum, tak to znamená, že jsou zvýhodněny neurony s váhovými vektory s malou normou. Při řídkých vstupních vektorech to může vést k problémům. Ve fázi po naučení, kdy hledám pro vstupní vektory jejich reprezentaci v mapě pak může vyhrávat několik málo neuronů s malou normou bez ohledu na to, jestli skutečně dobře reprezentují vstupní vektor. Důvodem je právě převládající vliv členu $\|\mathbf{w}_i\|_2^2$ nad členem skalárního součinu $\mathbf{w}_i^T \mathbf{x}$.

Protože v mSOM používám pouze skalární součin, určuje výběr vítěze pouze člen $\mathbf{w}_i^T \mathbf{x}$, pro který hledám maximum. Je dobré si ale uvědomit, že i v tomto případě má na vítězství vliv norma váhového vektoru.

4.3. Receptivní pole

Další důležitou vlastností je možnost nastavení receptivních polí jednotlivých neuronů (viz 2.3.4). Nejdříve naznačím, jak reprezentuji receptivní pole. Ke každému neuronu i , kromě jeho váhového vektoru $\mathbf{w}_i \in \mathbb{R}^N$, přiřadím navíc vektor receptivního pole $\mathbf{r}\mathbf{f}_i \in \{0, 1\}^N$. Tento vektor pak určuje, na které taxely bude moct neuron reagovat a které bude ignorovat. Pokud $\mathbf{r}\mathbf{f}_{i,j} = 0$, pak neuron i nereaguje na taxel, který je ve vstupním vektoru reprezentován na j -tém místě. Naopak pokud $\mathbf{r}\mathbf{f}_{i,j} = 1$, může neuron i na taxel reprezentovaný j -tou složkou ve vstupním vektoru reagovat.

Nejdříve je potřeba modifikovat určování vítěze. Vítězný neuron j se standardně při vstupním vektoru \mathbf{x} hledá tak, že

$$j(\mathbf{x}) = \arg \max_i \mathbf{w}_i^T \mathbf{x}, \quad i = 1, 2, \dots, m.$$

Pokud nemají mít některé složky vstupního vektoru vliv na hodnotu skalárního součinu, lze to jednoduše zajistit tím, že

$$\mathbf{w}_i = \mathbf{w}_i .* \mathbf{r}\mathbf{f}_i,$$

kde $.*$ značí prvkové násobení (obdobně jako v Matlabu). Tím se vynulují odpovídající složky ve váhovém vektoru. Pak jsou složky ze vstupního vektoru, které nemají mít pro daný neuron vliv na hodnotu skalárního součinu vynásobeny nulami z váhového vektoru a jejich vliv na hodnotu skalárního součinu je tak eliminován.

Kroky důležité pro zajištění nastavení receptivních polí v implementaci probíhají v tomto pořadí:

1. Vytvoření pomocných vektorů $\mathbf{w}\mathbf{p}_i$ $i = 1, 2, \dots, m$ s vynulovanými složkami podle receptivních polí

$$\mathbf{w}\mathbf{p}_i = \mathbf{w}_i .* \mathbf{r}\mathbf{f}_i, \quad i = 1, 2, \dots, m,$$

kde $.*$ značí prvkové násobení (obdobně jako v Matlabu).

2. Vítězný neuron j při vstupním vektoru \mathbf{x} hledám jako

$$j(\mathbf{x}) = \arg \max_i \mathbf{w}_i^T \mathbf{x}, \quad i = 1, 2, \dots, m.$$

3. Vynulování složek váhových vektorů \mathbf{w}_i $i = 1, 2, \dots, m$ podle receptivních polí $\mathbf{r}\mathbf{f}_i$

$$\mathbf{w}_i = \mathbf{w}_i \cdot * \mathbf{r}\mathbf{f}_i, \quad i = 1, 2, \dots, m,$$

kde $\cdot *$ značí prvkové násobení (obdobně jako v Matlabu) probíhá až na úplný **závěr** učení.

Důvod vytvoření pomocných váhových vektorů bude vysvětlen později.

4.4. Normování váhového vektoru

Původní myšlenkou bylo normovat váhové vektory v každém cyklu učení po vynulování složek váhového vektoru mimo receptivní pole. Kroky důležité pro normování mají pak následující pořadí operací:

1. Vynulování složek váhových vektorů \mathbf{w}_i , $i = 1, 2, \dots, m$ podle příslušných vektorů receptivních polí $\mathbf{r}\mathbf{f}_i$,

$$\mathbf{w}_i = \mathbf{w}_i \cdot * \mathbf{r}\mathbf{f}_i, \quad i = 1, 2, \dots, m,$$

kde $\cdot *$ značí prvkové násobení (obdobně jako v Matlabu).

2. Znornování váhových vektorů

$$\mathbf{w}_i = \frac{1}{\|\mathbf{w}_i\|} \cdot * \mathbf{w}_i, \quad i = 1, 2, \dots, m,$$

kde $\cdot *$ značí prvkové násobení (obdobně jako v Matlabu).

Uvedené kroky se provádí v každém cyklu učení. Je zřejmé, že na konci učení má váhový vektor jednotkovou normu. To mělo zajistit při určování vítěze úplnou nezávislost na normě váhového vektoru.

Experimentálně se osvědčil trochu modifikovaný způsob normování. Konkrétně se normuje v každém cyklu učení, ale vynulování váhových vektorů podle receptivních polí se provede až na úplný závěr učení. Kroky důležité pro normování se provádí v tomto pořadí:

1. Znornování váhových vektorů

$$\mathbf{w}_i = \frac{1}{\|\mathbf{w}_i\|} \cdot * \mathbf{w}_i, \quad i = 1, 2, \dots, m,$$

kde $\cdot *$ značí prvkové násobení (obdobně jako v Matlabu) se provede v každém cyklu učení.

2. Vynulování složek váhových vektorů \mathbf{w}_i , $i = 1, 2, \dots, m$ podle příslušných vektorů receptivních polí $\mathbf{r}\mathbf{f}_i$,

$$\mathbf{w}_i = \mathbf{w}_i \cdot * \mathbf{r}\mathbf{f}_i, \quad i = 1, 2, \dots, m,$$

kde $\cdot *$ značí prvkové násobení (obdobně jako v Matlabu) se provede až na úplný **závěr** učení.

To znamená, že váhové vektory po naučení nemají obecně jednotkovou normu. Při určování vítěze se využívá pomocného váhového vektoru, ve kterém se vynulují složky mimo receptivní pole.

Vliv uvedeného způsobu normování je takový, že pokud u váhového vektoru pouze složky v receptivním poli mají nenulové hodnoty, tak se vynulováním složek mimo receptivní pole nic nezmění a váhový vektor bude mít i po naučení a během určování vítěze jednotkovou normu. Naopak bude-li mít váhový vektor hodně nenulových složek s vyšší hodnotou mimo receptivní pole, tak po naučení i během určování vítěze bude mít norma váhového vektoru podstatně menší hodnotu než jedna. Tím dochází k oslabení takových neuronů.

4.5. Algoritmus učení mSOM

Zde uvedu kompletní algoritmus učení mSOM. Nechť má mapa m neuronů a vstupní vektor má N složek.

1. Náhodná inicializace váhových vektorů. Hodnota $n = 0$.
2. Náhodný výběr vstupního vektoru \mathbf{x} z trénovací množiny. Hodnota $n = n + 1$.
3. Vytvoření pomocných váhových vektorů $\mathbf{w}\mathbf{p}_i$, $i = 1, 2, \dots, m$ z váhových vektorů \mathbf{w}_i , $i = 1, 2, \dots, m$. Pomocné váhové vektory se liší od váhových vektorů vynulováním některých složek podle příslušných vektorů receptivních polí $\mathbf{r}\mathbf{f}_i$, $i = 1, 2, \dots, m$,

$$\mathbf{w}\mathbf{p}_i = \mathbf{w}_i \cdot * \mathbf{r}\mathbf{f}_i, \quad i = 1, 2, \dots, m,$$

kde $\cdot *$ značí prvkové násobení (obdobně jako v Matlabu).

4. Určení vítězného neuronu j pro vybraný vstupní vektor \mathbf{x}

$$j(\mathbf{x}) = \arg \max_i \mathbf{w}\mathbf{p}_i^T \mathbf{x}, \quad i = 1, 2, \dots, m.$$

5. Adaptace vah

$$\mathbf{w}_i(n+1) = \mathbf{w}_i(n) + \alpha(n)h(i, j(\mathbf{x}), n)(\mathbf{x} - \mathbf{w}_i(n)), \quad i = 1, 2, \dots, m,$$

kde $n \in \mathbb{N}$ představuje n -tý výběr vstupního vektoru z trénovací množiny, $\alpha(n)$ je rychlost učení, $h(i, j(\mathbf{x}), n)$ je funkce okolí. Význam je obdobný významu uvedenému v 3.2.4.

6. Znornování váhových vektorů

$$\mathbf{w}_i = \frac{1}{\|\mathbf{w}_i\|} \cdot * \mathbf{w}_i, \quad i = 1, 2, \dots, m,$$

kde $\cdot *$ značí prvkové násobení (obdobně jako v Matlabu).

7. Pokud není splněna podmínka ukončení (např. $n < P$, kde P je počet vektorů v trénovací množině) skok na bod 2.
8. Vynulování složek váhových vektorů \mathbf{w}_i , $i = 1, 2, \dots, m$ podle příslušných vektorů receptivních polí $\mathbf{r}\mathbf{f}_i$,

$$\mathbf{w}_i = \mathbf{w}_i \cdot * \mathbf{r}\mathbf{f}_i, \quad i = 1, 2, \dots, m,$$

kde $\cdot *$ značí prvkové násobení (obdobně jako v Matlabu).

4.6. Určování více vítězů s využitím laterální inhibice

Mějme naučenou mSOM. Pak mohu chtít v případě více doteků ve vstupním vektoru určit více vítězných neuronů, kde každý vítězný neuron reprezentuje místo jednoho doteku. Pro tento účel je vhodné využít při určování vítězů princip laterální inhibice. Tento princip vysvětlím na následujícím ilustrativním příkladu.

Ilustrativní příklad

Pro ilustraci si vezmu případ, kdy ve vstupním vektoru \mathbf{x} jsou tři na kůži vzdálené doteky.

1. Pro \mathbf{x} seřadím neurony podle intenzity pálení a určím vítězný neuron. Neurony sousedící s tímto vítězným neuronem utlumím.
2. Seřadím opět neurony podle jejich intenzity pálení.
3. Protože neurony kolem prvního vítěze jsou utlumené, bude v seřazeném seznamu neuronů podle intenzity pálení velmi pravděpodobně na druhém místě neuron reprezentující druhý dotek na kůži. Neurony sousedící s druhým neuronem, v seznamu seřazeném podle intenzity pálení, utlumím.
4. Seřadím neurony podle jejich intenzity pálení.
5. Neurony kolem prvního a druhého vítěze jsou utlumené. Proto je pravděpodobné, že v seznamu neuronů seřazeném podle intenzity pálení, bude na třetím místě neuron reprezentující oblast kůže třetího doteku. Neurony kolem tohoto neuronu utlumím.

Tímto způsobem jsem získal takové pořadí pálicích neuronů, že každý z prvních třech neuronů reprezentuje jeden dotek. Navíc tyto neurony by měly pálit výrazně více, než ostatní neurony, a proto by mělo být možné je odlišit, i když dopředu nebudu znát počet doteků ve vstupním vektoru.

4.6.1. Počet utlumení

Z algoritmu popsaném v příkladě to vypadá, že je dopředu nutné znát počet doteků proto, abych mohl nastavit kolikrát se mají utlmit sousední neurony (dále budu označovat jako počet utlumení). V ilustrativním příkladě byl počet utlumení tři. Dále budu označovat neuron kolem kterého dochází k utlumení jako vítězný. Při vhodném nastavení způsobu tlumení sousedních neuronů vítězným neuronem není nutné nastavovat počet utlumení přesně stejný jako je počet doteků. Možným způsobem takové realizace je vhodné nastavení míry tlumení neuronů sousedících s vítězným podle intenzity pálení vítězného neuronu. Pokud vítězný neuron hodně pálí, sousední neurony se hodně utlumí. Pokud naopak pálí málo, sousední neurony se utlumí málo. Pak je možné nastavit počet utlumení třeba na deset (předpokládám méně než deset současných doteků).

Ilustrativní příklad - pokračování

Pokračuji v ilustrativním příkladu. To znamená, že proběhly utlumení kolem tří neuronů.

6. Seřadím neurony podle jejich intenzity pálení.
7. a) Čtvrtý neuron v seřazeném seznamu neuronů podle intenzity pálení je v blízkosti některého neuronu, který zajišťoval utlumení sousedních neuronů. Čtvrtý neuron ze seznamu musí být proto již poměrně silně utlumen, a proto pálí jen slabě. Protože utlumení okolních neuronů čtvrtého neuronu je závislé na tom, jak pálí

čtvrtý neuron (pálí slabě), nemůže dojít k tomu, že by výrazně utlumil neuron, který dříve zajišťoval utlumení svých sousedů.

7. b) Pokud je čtvrtý nejvíce pálicí neuron mimo okolí neuronů, které dříve zajišťovaly utlumení sousedních neuronů, tak se nic špatného nemůže stát. V takovém případě je docela pravděpodobné, že počet současných doteků na kůži byl čtyři či více.

Tímto způsobem je možné provést větší počet utlumení, než je očekávaný maximální počet současných doteků a nemělo by dojít k výraznému utlumení neuronů, které skutečně reprezentují jednotlivé doteky ve vstupním vektoru.

4.6.2. Nastavení hodnot tlumení

Aby prezentovaný princip skutečně fungoval, je nutné vhodně nastavit konkrétní hodnoty míry tlumení sousedních neuronů podle pálení vítězného neuronu. Způsob, který jsem vyzkoušel a dobře fungoval je následující:

- Pokud je neuron i ve vzdálenosti $[0.95, 1.5]$ od vítězného neuronu j , který pálí intenzitou d_j , pálení neuronu i po utlumení je $d_i = \max\{d_i - 0.8d_j, 0\}$.
- Pokud je neuron i ve vzdálenosti $(1.5, 3.5]$ od vítězného neuronu j , který pálí intenzitou d_j , pálení neuronu i po utlumení je $d_i = \max\{d_i - 0.6d_j, 0\}$.

V uvedeném způsobu tlumení si lze všimnout maxima. To je proto, aby intenzita pálení neuronu po utlumení nemohla být záporná.

4.6.3. Shrnutí algoritmu určování více vítězů s laterální inhibicí

Na závěr uvedu celý algoritmus:

1. Nastav počet utlumení k (např. $k = 10$ - maximální možný počet současných doteků na kůži), $i = 0$.
2. Při daném vstupním vektoru \mathbf{x} urči intenzitu pálení všech neuronů a neurony podle intenzity pálení seřaď (sestupně).
3. Nastav $i = i + 1$. V seřazeném seznamu neuronů podle intenzity pálení vyber i -tý neuron v pořadí a utlum podle intenzity pálení tohoto neuronu jeho sousední neurony.
4. Seřaď neurony podle intenzity pálení.
5. Pokud $i < k$ skoč na krok 3.

V této práci jsem chtěl pouze ukázat, že princip laterální inhibice při určování více vítězů funguje. To se podle provedených experimentů povedlo dobře (bude později prezentováno na konkrétním příkladu). Nicméně v rámci této práce už nezbyl prostor pro detailnější zkoumání tohoto tématu. Proto je možné, že v některých mezních či speciálních případech, které se nestihly pořádně prozkoumat, nebude uvedený způsob laterální inhibice dávat optimální výsledky a bude vhodné provést určitou modifikaci.

5. Hodnocení naučených map

V této kapitole rozeberu způsob hodnocení naučených map. Hodnocení odráží splnění požadavků na vlastnosti map, které jsou uvedeny v 2.3. V mé práci se nejvíce zaměřím na vlastnost, kterou považuji za nejdůležitější, zachování topologie taxelů ze vstupní vrstvy v jejich reprezentaci ve výstupní vrstvě.

Nejdříve popíši způsob vizualizace map a vektorů vah neuronů. Následně se zmíním o hodnocení pálení neuronů při daném vstupním vektoru. Pak se budu věnovat hodnocení zachování topologie a hodnocení vlivu četnosti aktivace taxelu na jeho zastoupení v mapě. Na závěr uvedu způsob hodnocení naučených map, který budu používat.

Všechny funkce uvedené v této kapitole lze nalézt na přiloženém CD.

5.1. Grafické znázornění

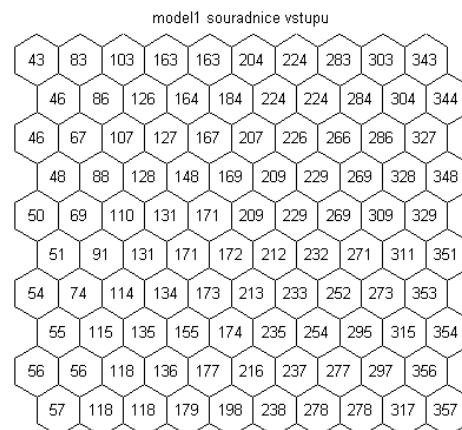
V této podkapitole uvedu způsob grafického znázornění mapy a vektoru vah neuronu.

5.1.1. Grafické znázornění mapy

Mapu budu graficky znázorňovat tak, že každému neuronu v mřížce mapy přiřadím jeden taxel. Taxelů silněji "propojených" s jedním daným neuronem je většinou více. Snaha je přiřadit takový taxel, který daný neuron nejlépe reprezentuje. Osvědčilo se volit taxel, se kterým je daný neuron nejsilněji propojen. To znamená zvolit taxel odpovídající složce o maximální hodnotě ve váhovém vektoru daného neuronu.

Ilustrativní příklad

Příklad uvedeného znázorněné mapy je na obr. 5. K vykreslení lze použít funkci `seeMapF2`.

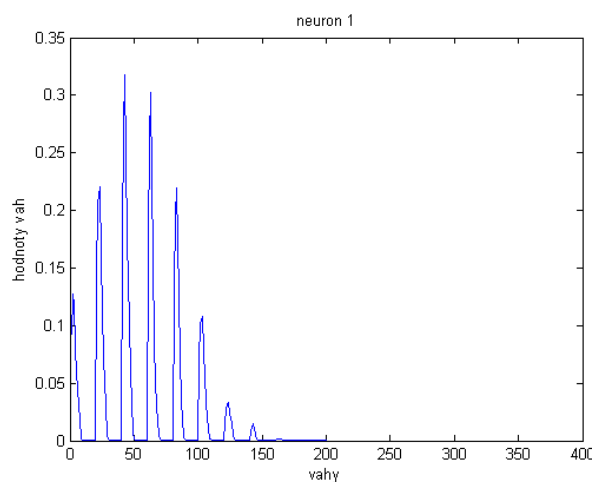


SOM 14-Mar-2014

Obrázek 5. Přiřazení taxelů k neuronům

5. Hodnocení naučených map

Na obr. 6 se lze přesvědčit, že pro první neuron z obr. 5 (vlevo nahoře) má největší hodnotu 43. složka váhového vektoru tohoto neuronu.



Obrázek 6. Váhový vektor neuronu 1

Číslování neuronů v mapě

Číslování neuronů v grafickém znázornění mapy je takové, že čísla ve sloupcích rostou od shora dolů. To znamená, že na obr. 5 v prvním sloupci má neuron nejvíce nahoře číslo 1, neuron pod ním má číslo 2 atd. Po nejspodnějším neuronu ve sloupci číslování pokračuje neuronem nejvíce nahoře ve vedlejším sloupci vpravo. Na uvedeném obrázku to znamená, že neuron ve druhém sloupci nejvíce nahoře má číslo 11.

Obarvení neuronů v mapě

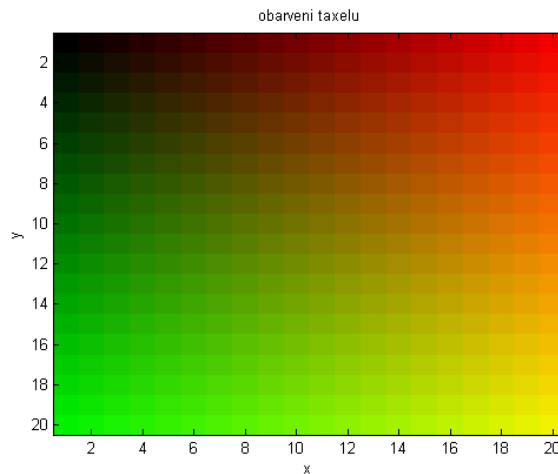
Ze samotného obr. 5 nelze poznat, zda sousedící neurony reprezentují i vzájemně blízké taxely. Abych to mohl poznat, musím navíc znát pozice taxelů v prostoru. Pro zvýšení názornosti je pak možné obarvit neurony v mapě podle pozice taxelů tak, aby neurony reprezentující blízké taxely měly podobnou barvu. To lze realizovat tak, že každému taxelu přiřadím podle jeho pozice určitou barvu. Konkrétně se mohou souřadnice pozic taxelů normalizovat a pak se každému taxelu podle jeho pozice přiřadí barva v barevném modelu RGB, kde normované souřadnice pozice taxelů představují barevné složky modelu RGB. Obarvení taxelů, podle jejich souřadnic pro čtvercovou kůži 20×20 s celočíselnými pozicemi taxelů, je znázorněno na obr. 7.

Mapa z obr. 5 po obarvení neuronů, podle pozice taxelů, které reprezentují, je znázorněna na obr. 8 vlevo.

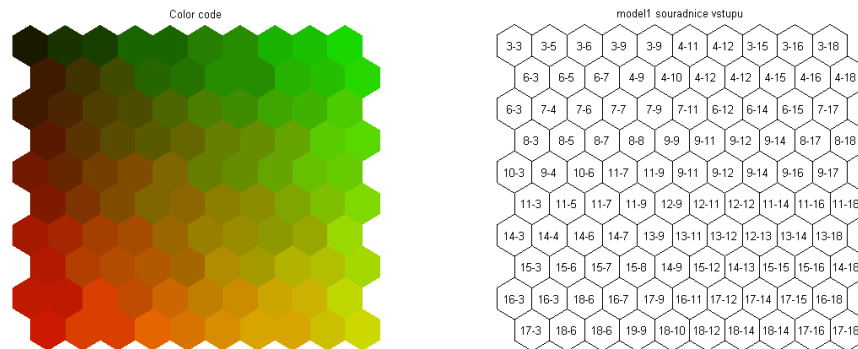
Vpravo na stejném obrázku jsou u jednotlivých neuronů pozice taxelů, které reprezentují. Pro vykreslení těchto dvou obrázků lze použít funkci `seeMapF`. Podle obarvení neuronů se dá rychle udělat představa o tom, jak mapa reprezentuje jednotlivé taxely a zda blízké neurony reprezentují i blízké taxely.

Zhodnocení použitého přiřazení

Podle mých zkušeností přiřazení taxelů neuronům podle maxima váhového vektoru dává velmi dobrou představu o naučení mapy. Je dobré si ale uvědomit, že teoreticky se



Obrázek 7. Obarvení taxelů podle jejich pozice



SOM 14-Mar-2014

Obrázek 8. Obarvení neuronů podle pozice reprezentovaných taxelů (vlevo), souřadnice pozic taxelů reprezentovaných neuronů (vpravo)

nemusí tímto způsobem odhalit některé chyby, a proto je vhodné (alespoň při odlišných simulacích než byly dosud provedeny) zkontrolovat raději i několik váhových vektorů jednotlivých neuronů.

Obarvení neuronů v mapě při velké kůži

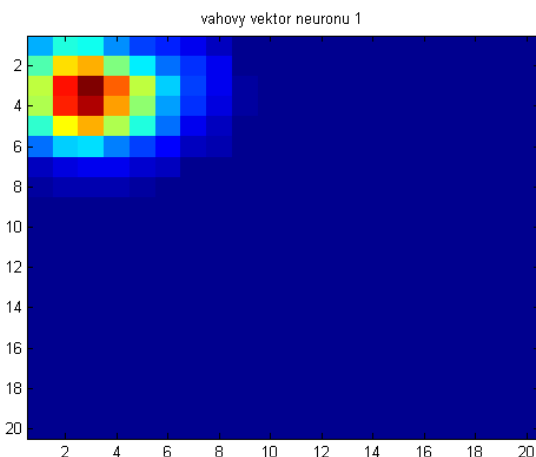
Pokud jsou rozměry kůže velké (např. celá kůže iCuba) a taxely jsou blízko sebe, může se stát, že z lokálního pohledu vzdálené taxely budou mít podobnou barvu. To je dáno tím, že barevné složky RGB modelu jsou normované, takže když musí daná barevná složka (vždy se stejným rozsahem) pokrýt velkou vzdálenost, snižuje se její rozlišovací schopnost. Rozlišovací schopnost může být pak tak malá, že i lokálně vzdálené taxely budou mít podobnou barvu. V takovém případě není vhodné použít obarvení taxelů pro celou kůži. Lze však použít funkci `seeMapF3`, která umožní nastavit pouze obarvení

vybraného obdélníkového výseku kůže. Neurony mimo vybraný obdélník mají v grafickém znázornění mapy bílou barvu. Zmíněná funkce vykreslí kromě obarvené mapy i mapu s přiřazením čísel taxelů neuronům (obdobně jako funkce `seeMapF2`).

5.1.2. Grafické znázornění vektoru vah neuronu

Grafické znázornění celé mapy neobsahuje informace o váhových vektorech (s výjimkou `maxim`) neuronů v mapě. Znázornění váhových vektorů může být velmi užitečné při analýze naučených map a při odhalování příčin špatného učení map. Proto je nutné vizualizovat i váhové vektory. Přírozeným způsobem je vynést na osu x očíslování složek váhového vektoru a na osu y vynést hodnoty jednotlivých složek. Takové znázornění bylo použito na obr. 6.

V případě obdélníkové kůže, s celočíselnými pozicemi taxelů, budu vyjadřovat váhový vektor také v ploše. To znamená, že se složky váhového vektoru přeskládají do obdélníkové matice o rozměrech kůže. Souřadnice prvků matice korespondují s pozicemi taxelů. Pak je možné vizualizovat matici tak, že se prvkům matice podle jejich hodnoty přiřadí barva. Pro váhový vektor z obr. 6 je tímto způsobem provedena vizualizace na obr. 9. Z obr. 9 je přímo vidět, jak silně je neuron propojen s jednotlivými taxely na



Obrázek 9. Znázornění váhového vektoru v rovině

kůži. Pro vykreslení lze použít funkci `vahyNeuronu`.

5.2. Hodnocení pálení neuronů v mapě

Nejdříve je nutné si uvědomit, co přesně znamená, že neuron reprezentuje nějakou část kůže. To znamená, že neuron pálí při stimulaci této části kůže. Pro přehledné znázornění a snadný popis je pak výhodné každému neuronu přiřadit pouze jeden taxel, který by měl být přibližně ve středu oblasti kůže, při jejíž stimulaci neuron pálí. Adeptem na takové přiřazení je přiřazení, používané ke grafickému znázornění, popsané v 5.1.1. Uvedený způsob však sám o sobě nezajišťuje, že neurony budou skutečně silně pálit při doteku v oblasti jim přiřazeného taxelu. Pokud by v mapě například existoval jeden neuron s maximální možnou hodnotou všech vah, tak by reagoval nejsilněji jak jen to je možné, na všechny existující doteky a nedalo by se korektně hovořit o tom, že ostatní neurony reprezentují taxely uvedené v grafickém znázornění mapy. Všechny taxely by nejlépe reprezentoval (ve smyslu výše uvedeném) pouze jeden silný neuron.

Proto je nutné ověřit, že při stimulaci několika taxelů skutečně pálí neurony, které mají v grafickém znázornění mapy (viz 5.1.1) přiřazeny stimulované taxely.

5.2.1. Jeden současný dotek

Pro vstupní vektory s jedním dotekem budu uvedenou korespondenci ověřovat jednoduše tak, že dám na vstup mapy vektor s dotekem menší velikosti, s přibližným středem v určitém taxelu a zjistím, zda nejvíce pálí některý z neuronů, který podle grafického znázornění mapy reprezentuje taxely blízko taxelu středu doteku.

Experimentálně se ukázalo, že tato korespondence je většinou v pořádku (pokud neuvažuji na první pohled chybně naučené mapy). Přesto je vhodné preventivně tuto korespondenci, zvláště u simulací s neprozkoumanými parametry, kontrolovat. V simulaci v 6.2.2 je ukázka splnění této korespondence.

5.2.2. Více současných doteků

Důležitou požadovanou vlastností map je schopnost poradit si i s více současnými doteky, které jsou od sebe na kůži vzdáleny (dále multidoteky). U naučené mapy to znamená, že při multidoteku bude pálit silně více neuronů, kde každý neuron reprezentuje jednu ze stimulovaných oblastí kůže (dotek). Pro tento účel je vhodné použít určování vítězů s laterální inhibicí.

Splnění této vlastnosti budu ověřovat podobně jako u vstupních vektorů s jedním dotekem. Pro každý dotek ze vstupního vektoru musím najít podle přiřazení taxelů neuronům v grafickém znázornění jemu odpovídající neuron a pak zkontrolovat, zda tento neuron (či nějaký jemu blízký) je mezi několika neurony, které pálí s velkou intenzitou.

Protože podle provedených simulací u na pohled dobře naučených map podle grafického znázornění funguje určování více vítězů při multidotecích dobře, nebudu to při každé simulaci kontrolovat. V simulaci v 6.2.2 je ukázáno určení více vítězů s laterální inhibicí.

5.3. Hodnocení zachování topologie

Zachování topologie vstupní vrstvy ve výstupní vrstvě považuji v této práci za nejdůležitější vlastnost map. Tomu by měl být přizpůsoben i sběr trénovacích doteků (např. zabránění častým koaktivacím některých vzájemně vzdálených taxelů). Pak by nemělo být příliš obtížné tuto vlastnost zkoumat.

5.3.1. Grafická metoda

V 5.2 bylo řečeno, že grafické znázornění mapy uvedené v 5.1.1 dává dobrou představu o tom, jaké oblasti kůže jednotlivé neurony reprezentují. Proto lze použít uvedené grafické znázornění mapy ke zkoumání zachování topologie. V případě obarvení neuronů je velmi dobře vidět, zda se topologie zachovává. Konkrétně mají-li sousední neurony podobné barvy, pak reprezentují i blízké taxely.

5.3.2. Číselná míra

Hodnocení zachování topologie vstupní vrstvy ve výstupní vrstvě číselnou mírou, představuje pomocný způsob hodnocení. Tato míra neslouží jako náhrada komplexnějšího hodnocení založeného hlavně na grafickém znázornění. Slouží jako nástroj pro přehledné

5. Hodnocení naučených map

porovnání většího počtu naučených map s blízkými parametry. Hodnocení naučených map, pouze na základě této číselné míry, nemusí ve všech existujících případech dávat správné výsledky. Proto není vhodné používat k hodnocení pouze číselnou míru. Funkci, která implementuje tuto míru, lze nalézt pod názvem `qmSOMF` na příloženém CD.

Požadavky na testovací množinu

Míra vyžaduje testovací množinu doteků. Tato množina by měla obsahovat ideálně pouze vektory, vždy s jedním souvislým dotekem (jednodotek) rozumné velikosti (je stimulováno zároveň několik velmi blízkých taxelů). Míra předpokládá, že taxely aktivované během jednoho doteku, jsou blízko sebe.

Algoritmus výpočtu

1. Nechtě $M = \emptyset, V = \emptyset$.
2. Vezmi vstupní vektor \mathbf{x} z testovací množiny T a odeber ho z této množiny.
3. Do množiny M ulož indexy složek vektoru \mathbf{x} s hodnotou 1.
4. Vytvoř pomocné vektory $\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_m$, $m = |M|$ tak, že pomocné vektory jsou po dvou různé a každý pomocný vektor má pouze jednu složku rovnou 1 a další složky jsou rovné 0. Nenulové jsou pouze složky $i \in M$.
5. Pro každé dva pomocné vektory $\mathbf{p}_a, \mathbf{p}_b$, $a \in \{1, 2, \dots, m\}$, $b \in \{1, 2, \dots, m\}$, $a \neq b$ najdi jejich vítězné neurony v mapě a urči jejich vzdálenost v mřížce a přidej tuto vzdálenost do multimnožiny V .
6. Smaž pomocné vektory $\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_m$, $m = |M|$, odeber všechny prvky z množiny M .
7. Pokud $T \neq \emptyset$ skoč na bod 2.
8. Spočítej průměrnou hodnotu $prum$ a směrodatnou odchylku std prvků multimnožiny V .
9. Vrať hodnotu $prum$ a std .

Interpretace a omezení míry

Míra vyjadřuje, jak jsou taxely, které jsou společně aktivované během doteku, daleko od sebe reprezentované na mapě. Samotným pojmem míra budu označovat průměrnou hodnotu $prum$. Pokud budu mít na mysli směrodatnou odchylku std speciálně to uvedu.

Mapa, která zachovává topologii by měla mít hodnotu míry menší než mapa, která nezachovává topologii. Hodnota míry je závislá na množině testovacích doteků. S velikostí doteků roste i velikost hodnoty míry. Pokud nebude většinou platit, že taxely aktivované během jednoho doteku jsou blízko sebe, nebude míra dávat dobré výsledky. Hodnota míry je také závislá na parametrech učení a mapy. Například mapa naučená na trénovací množině s jednodoteky může mít v případě nezachování topologie menší hodnotu míry, než mapa naučená na multidotekách, která zachovává topologii. Proto je vhodné používat ukazatel hlavně k porovnání map naučených při stejných parametrech. Pokud to nedodržíme, je vhodné nejdříve porovnáním výsledků míry a grafické metody určit, v jakém intervalu lze výsledkům míry s velkou spolehlivostí důvěřovat a v jakém intervalu nelze udělat z hodnoty míry spolehlivý závěr o zachování topologie mapy. Při rozhodování nám může pomoci i hodnota standardní odchylky, která může odhalit některé problémy, které nejsou pouze z hodnoty průměru vidět.

Možnost nastavení prahu pálení

V implementované verzi funkce `qmSOMF` je navíc možné nastavit práh pálení pro započítání neuronů do hodnocení. To znamená, že je možné touto mírou hodnotit i mapy, kde jsou např. krajní taxely špatně reprezentované (vlivem hraničního efektu (viz 3.4) a menšího počtu stimulací). Při vhodném nastavení prahu pálení pak nemají tyto slabě reprezentované taxely na hodnotu míry vliv. V některých případech to může být žádoucí, v některých naopak ne. Standardně je hodnota prahu přednastavena na 0.07.

5.4. Hodnocení vlivu četnosti aktivace taxelu na jeho zastoupení v mapě

Hodnocením vlivu četnosti aktivace taxelu na jeho zastoupení v mapě se rozumí hodnocení splnění vlastnosti popsané podrobně v 2.3.2. Stručně řečeno jde o to, zda častěji aktivované části kůže zabírají více místa na mapě, než ty méně často aktivované části kůže. Splnění této vlastnosti lze snadno pozorovat v grafickém znázornění mapy uvedeném v 5.1.1.

Aby se projevila tato vlastnost je nutné, aby rozložení aktivace taxelů v trénovací množině nebylo rovnoměrné. To pro zkoumání ostatních vlastností není příliš žádoucí, proto se o to v této práci nebudu speciálně pokoušet. Pro ilustraci, že navržená neuronová síť tuto vlastnost splňuje jsem provedl simulaci v 6.2.5. Simulací, kde se tato vlastnost projevila, jsem provedl více, ale protože se v této práci zaměřuji hlavně na zkoumání vlastnosti zachování topologie, nebudu je zvlášť uvádět.

5.5. Způsob hodnocení map v této práci

Nejdříve bych chtěl zdůraznit, že při hodnocení map s dosud nevyzkoušenými parametry je vhodné pojmout hodnocení komplexně. To znamená, zhodnotit grafické znázornění celé mapy, vektory vah několika neuronů, vyzkoušet zda na doteky skutečně reagují správné neurony atd. Popisovat takové komplexní hodnocení map v této práci není kvůli jejímu rozsahu možné. Proto budu u jednotlivých naučených map uvádět a používat k hodnocení často jen grafické znázornění celé mapy (viz 5.1.1). Hlavní důvody jsou:

- Nejlépe znázorňuje topologii vstupní vrstvy ve výstupní vrstvě,
- zpravidla pokud je toto grafické znázornění mapy v pořádku (zachování topologie), jsou v pořádku i další požadované vlastnosti mapy,
- žádný jiný uvedený způsob znázornění nedává na takovém prostoru tolik informací o mapě.

Dále je potřeba upozornit, že učení není deterministické (náhodná inicializace vah, náhodný výběr vektorů z trénovací množiny), proto je potřeba jednu simulaci víckrát opakovat. Uvádět grafické znázornění map ze všech opakování není v rozsahu této práce možné. Proto uvedu zpravidla jediné a dále uvedu kolik map z provedených opakování splňovalo vlastnost zachování topologie (na základě uvedeného grafického znázornění map). Tato vlastnost zpravidla velmi spolehlivě zajišťuje splnění dalších požadovaných vlastností. Pro ilustraci číselné míry popsané v 5.3.2 uvedu u některých simulací hodnoty této číselné míry.

To, na co se během hodnocení naučených map zaměřím, se také bude lišit podle toho, co budu chtít na dané mapě prezentovat.

6. Testování naučených neuronových sítí a interpretace výsledků

V této kapitole se věnuji testování navrženého modelu neuronové sítě pro reprezentaci taktilních senzorů. Pro dobré pochopení vlastností neuronové sítě a ověření splnění požadovaných vlastností (viz 2.3) bylo nutné provést celou řadu simulací. Učení neuronové sítě bylo provedeno nejdříve na počítačem generovaných dotecích (umělé doteky), následně na dotecích z kůže robota iCub. Vzhledem k omezenému prostoru této práce a velkému počtu provedených simulací, uvedu zde pouze některé simulace, na kterých budu prezentovat vlastnosti naučených map.

U každé simulace nejdříve uvedu její důležité parametry. Po jedné nebo několika podobných simulacích provedu zhodnocení a interpretaci výsledků simulací.

Každou z uvedených map lze nalézt na příloženém CD (podle jejího označení, např. 1403p2o4). Pro každou z uvedených simulací lze nalézt na příloženém CD skript, který danou simulaci spouští a všechny funkce, které používal. Příložené CD dále obsahuje všechny další soubory (např. obrázky ve formátu `fig`), skripty a funkce, které uvedu v textu.

Nejdříve v této kapitole popíšu význam parametrů simulací. Následně se budu věnovat učení s umělými doteky. Poté se budu zabývat učením s doteky z robota iCub.

6.1. Parametry simulací

Jako přehledný, poměrně komplexní, stručný a rychlý způsob vyjádření parametrů doteků, učení a map se ukázalo uvádět u každé simulace hodnoty parametrů v této formě:

```
m=20; n=20; o=10; p=10; %rozmery kuze(m,n)/mapy(o,p)
pocTrDot=20000; %pocet trenovacich doteku
dotykPst=1; %pravdepodobnost aktivace nestredoveho taxelu doteku
pocTrEp=6; %pocet trenovacich epoch-parametr trainlen
radiusIni=3; radiusFin=1; %radius init/final
RF=ones(o*p,m*n); RF(51:100,1:120)=0; %receptive fields
RF(1:50,281:400)=0;
```

V případě učení z dat z robota iCub není potřeba uvádět parametry související s generováním umělých doteků.

Nejdříve proto vysvětlím význam všech uvedených parametrů.

6.1.1. Parametry umělých doteků

Mějme obdélníkovou kůži o rozměrech $m * n$ taxelů. Na této kůži se dotek generuje tak, že se náhodně vybere taxel, který pak představuje střed doteku. Tento taxel se aktivuje. Podle hodnoty parametru `dotykPst` se aktivuje několik dalších taxelů kolem středu doteku. V tento okamžik je stav jednotlivých taxelů reprezentován maticí o rozměrech kůže. Hodnoty prvků jsou 1 (taxel aktivován) a 0 (taxel neaktivován). Z této matice se vstupní vektor získá tak, že se jednotlivé řádky této matice poskládají za sebe

tímto způsobem (*řádek1, řádek2, ..., řádek m*). Očíslování taxelů odpovídá pozici taxelu ve vstupním vektoru. To znamená, že v prvním řádku obdélníkové kůže jsou taxely očíslované zleva směrem doprava hodnotami 1 až n , ve druhém řádku hodnotami $n + 1$ až $2n$ atd.

Matice, která představuje trénovací množinu, má v řádcích vstupní vektory. Počet řádků této matice odpovídá počtu vstupních vektorů a budu ho označovat jako `pocTrDot`.

Parametr `dotykPst`

Nyní vysvětlím význam parametru `dotykPst`. Po vygenerování středu doteku (u něj se vždy nastaví jednička) se okolní taxely (je jich 8 kromě okrajů kůže) aktivují s pravděpodobností uvedenou v parametru `dotykPst`. To znamená, že je-li hodnota tohoto parametru 1 aktivuje se $1 + 1 * 8 = 9$ taxelů (pokud to není na kraji). Dotek na kůži pak vypadá zhruba takto:

```
1 1 1
1 1 1
1 1 1
```

Pokud je hodnota tohoto parametru 0.2 pak se aktivuje v průměru $1 + 0.2 * 8 = 2.6$ taxelů. Rozložení pravděpodobnosti aktivace taxelů v doteku je:

```
0.2 0.2 0.2
0.2 1 0.2
0.2 0.2 0.2
```

To znamená, že průměrný dotek bude vypadat na kůži třeba nějak takto:

```
0 0 1
0 1 0
0 1 0
```

6.1.2. Parametry `map`

Parametry společné všem mapám

Zde uvedu hodnoty parametrů společné všem mapám v této práci. Protože se jedná o standardní parametry SOM toolboxu odkážu čtenáře buď na stručnější popis [9], nebo na podrobnější popis [12].

```
som_topol.lattice=hexa;
som_topol.shape=sheet;
%som_seqtrain--sequential learning
neigh=gaussian; %function of radius
%radius goes linearly from radius_ini to radius_fin
alpha_type=inv;
alpha_ini=0.5;
```

Jen stručně uvedu, že trénování je vždy sekvenční, tak jak je uvedeno v 4. V SOM toolboxu `radius` souvisí s funkcí okolí a `alpha` souvisí s funkcí rychlosti učení. Význam funkce okolí a funkce rychlosti učení je vysvětlen v 3.2. Za zmínku ještě stojí, že hodnota `radius` během učení od počáteční hodnoty ke konečné hodnotě lineárně klesá.

Proměnné parametry

Mějme mapu o velikosti $o * p$ neuronů. Parametr RF určuje nastavení receptivních polí (viz 4.3). Parametr je matice o rozměrech matice váhových vektorů `sMap.codebook`. Každý řádek představuje jeden neuron. Sloupce představují jednotlivé taxely. Pokud prvek matice RF má hodnotu 1 znamená to, že příslušný neuron je během učení citlivý na daný taxel. V případě hodnoty prvku 0 příslušný neuron není citlivý na daný taxel. Proměnná `pocTrEp` představuje počet trénovacích epoch, proměnné `radiusIni` a `radiusFin` představují počáteční a konečnou hodnotu proměnné `radius`. Pro podrobnosti čtenáře opět odkáži na [12].

6.2. Učení s umělými doteky

V rámci této práce bylo provedeno několik stovek simulací. Ve většině provedených simulací byly doteky generované počítačem (umělé doteky). Cílem bylo provést takové simulace, aby se ukázalo, zda je možné použít navrženou neuronovou síť pro reprezentaci kůže u robota iCub a jak je nutné nastavit parametry pro úspěšné učení. Postupoval jsem postupně od jednoduchých simulací, které byly značně vzdálené od situace na iCubovi, až ke složitějším simulacím, které by měly být velmi blízké situaci na iCubovi. To přibližně znamená, že jsem postupně zvětšoval velikost kůže a map, ale také jsem přidával počet současných stimulací na kůži a experimentoval s dalšími parametry. Vždy když jsem zvládl určitý případ a pochopil, co musí být splněno pro úspěšné učení v takovém případě, tak jsem si vzal o něco komplikovanější případ a zase na něm pracoval, dokud se mi ho nepodařilo vyřešit a pochopit, co musí být splněno pro úspěšné učení. To znamená, že jsem systematicky postupoval tak, abych se co nejrychleji dostal co nejblíže situaci na iCubovi.

Provádění simulací je časově poměrně náročné a je nutné simulace opakovat (pět opakování jedné simulace pak trvalo v některých případech více než jeden den), proto bylo nutné vážít provedení každých několika simulací. Nebylo proto možné postupovat tak, že bych si vybral některý z parametrů, který bych postupně měnil a sledoval vliv na kvalitu učení. Parametrů učení bylo poměrně dost a než bych se tímto způsobem propracoval k situaci, která je na iCubovi, musel bych provést mnohonásobně více simulací než bylo možné v daném čase provést.

Z uvedených důvodů jsem prováděl poměrně detailní analýzu naučených map, abych dokázal rychle určit, které parametry je potřeba změnit pro dosažení lepší úspěšnosti a porozuměl učení. Zároveň jsem se snažil provést simulace, které mi měly potvrdit správnost mých hypotéz důvodů, proč se mapy špatně učily.

Použitý způsob práce mi umožnil dostat se poměrně rychle k simulacím podobným situaci na iCubovi a udělat si určitou představu o tom, jak se mají nastavit parametry učení. Na druhou stranu jsem nezískal tak systematické pochopení vlivu jednotlivých parametrů na učení jako kdybych si mohl dovolit měnit postupně jednotlivé parametry a získat závislosti výsledků učení na těchto parametrech.

Uvedený způsob postupu od jednoduchých případů až k těm složitějším (iCub), na základě detailní analýzy naučených map, je problematické v rozsahu této práce přesně popsat. Musel jsem vybrat pouze ukázky některých simulací, na kterých budu prezentovat splnění požadovaných vlastností neuronové sítě pro reprezentaci taktilních senzorů (viz 2.3) a některé jejich důležité vlastnosti.

Simulace s podobným cílem jsem sdružil do jedné série. Na závěr každé série je uvedena interpretace a zhodnocení výsledků simulací dané série.

6.2.1. Přehled sérií simulací

Pro lepší orientaci v provedených sériích simulací, uvedu přehled simulací s jejich krátkým popisem.

1. série V této sérii budu zjišťovat, jestli na dotek na vstupu pálí neuron, který bych očekával podle grafického znázornění mapy s přiřazením taxelů neuronům. Dále vyzkouším určování více vítězů s principem laterální inhibice.
2. série V této sérii budu zjišťovat, zda je neuronová síť schopna se učit při různých poměrech velikosti mapy (počtu neuronů) a velikosti kůže (počtu taxelů).
3. série V této sérii budu zkoumat, zda se mapa dobře naučí na vstupních vektorech se dvěma současnými doteky. Dále vyzkouším, zda míra pro hodnocení zachování topologie umožňuje rozlišit, zda mapa splňuje uvedenou vlastnost.
4. série V této sérii budu výrazně více stimulovat určitou oblast kůže a budu chtít zjistit, jak se to projeví v naučené mapě.
5. série V této sérii budu zkoumat, zda se mapa dobře naučí na vstupních vektorech se šesti současnými doteky. Dále předvedu vliv velikosti doteků (počtu aktivovaných taxelů) a vliv velikosti trénovací množiny na učení.
6. série V této sérii předvedu učení mapy na větší kůži. Dále se budu zabývat vhodným nastavením receptivních polí a vlivem počtu neuronů v mapě na učení.

6.2.2. 1. série simulací: pálení neuronů na doteky

V této sérii budu zjišťovat, jestli na dotek na vstupu pálí neuron, který bych očekával podle grafického znázornění mapy s přiřazením taxelů neuronům. Dále vyzkouším určování více vítězů s principem laterální inhibice.

1. simulace

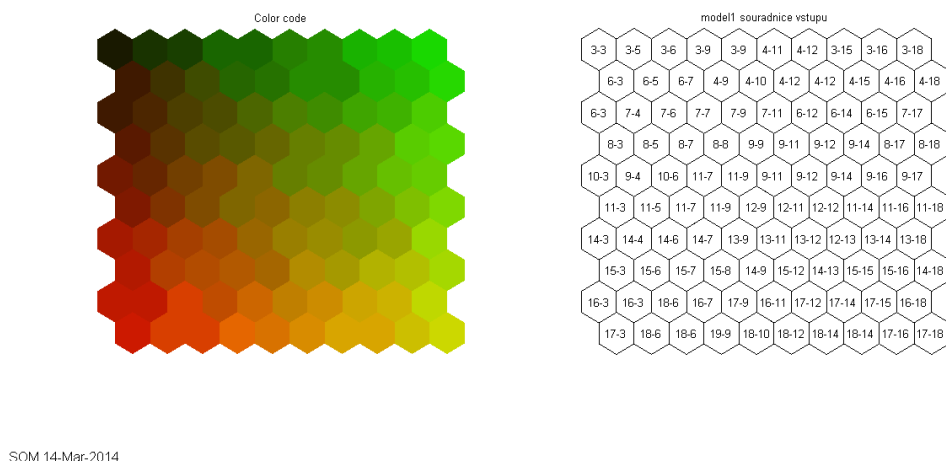
Mapa má označení 1403p2. Parametry simulace jsou:

```
m=20; n=20; o=10; p=10; %rozmary kuze(m,n)/mapy(o,p)
%jednodoteky
pocTrDot=20000; %pocet trenovacich doteku
dotykPst=1; %pravdepodobnost aktivace nestredoveho taxelu doteku
pocTrEp=6; %pocet trenovacich epoch-parametr trainlen
radiusIni=3; radiusFin=1; %radius init/final
RF=ones(o*p,m*n); RF(51:100,1:200)=0; %receptive fields
RF(1:50,201:400)=0;
```

Grafické znázornění jedné z takto naučených map (1403p2o5) je na obr. 10. Z pěti provedených simulací ve třech případech splňovala mapa požadovanou vlastnost zachování topologie.

Na vstupní vektor \mathbf{x}_1 s dotekem se středem o souřadnicích (5, 7) na kůži a parametrem $dotykPst = 1$ reaguje nejsilněji (funkce `som_bmusDP`) neuron 22. Na obr. 10 je tomuto neuronu přiřazen taxel o souřadnicích (6, 7). To je velmi blízko středu doteku ze vstupního vektoru.

Na vstupní vektor \mathbf{x}_2 s dotekem se středem o souřadnicích (12, 10) a parametrem $dotykPst = 1$ reaguje nejsilněji neuron 46. Na obr. 10 je tomuto neuronu přiřazen taxel o souřadnicích (12, 9). To je velmi blízko středu doteku ze vstupního vektoru.



Obrázek 10. Simulace 1, vlevo jsou neurony obarveny podle pozice jim přiřazených taxelů, vpravo jsou u neuronů uvedeny souřadnice jim přiřazených taxelů

Na vstupní vektor \mathbf{x}_3 s dotekem se středem o souřadnicích $(7, 4)$ a parametrem $dotykPst = 1$ reaguje nejsilněji neuron 13. Na obr. 10 je tomuto neuronu přiřazen taxel o souřadnicích $(7, 4)$. To přímo odpovídá středu doteku ze vstupního vektoru.

Na vstupní vektor \mathbf{x}_4 s dotekem se středem o souřadnicích $(16, 15)$ a parametrem $dotykPst = 1$ reaguje nejsilněji neuron 89. Na obr. 10 je tomuto neuronu přiřazen taxel o souřadnicích $(17, 15)$. To je velmi blízko středu doteku ze vstupního vektoru.

Na vstupní vektor \mathbf{x}_5 s dotekem se středem o souřadnicích $(1, 1)$ a parametrem $dotykPst = 1$ reaguje nejsilněji neuron 1. Na obr. 10 je tomuto neuronu přiřazen taxel o souřadnicích $(3, 3)$. Taxel reprezentovaný neuronem 1 je ke středu doteku nejbližší ze všech taxelů znázorněných v mapě.

Na vstupní vektor \mathbf{x}_6 s doteky z vektorů $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5$ využiji při hledání nejsilněji pálících neuronů princip laterální inhibice (funkce `som_bmusDPLatInh`). Intenzita pálení jednotlivých neuronů je na obr. 11.

Na obr. 11 je vidět, že nejvíce pálí neurony 13 (dotek v \mathbf{x}_3), 22 (dotek v \mathbf{x}_1), 46 (dotek v \mathbf{x}_2) a 89 (dotek v \mathbf{x}_4). Nejvíce pálící neurony odpovídají vítězným neuronům při samotných dotecích v $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4$. Chybí pouze neuron 1, který byl vítězným neuronem při doteku v \mathbf{x}_5 .

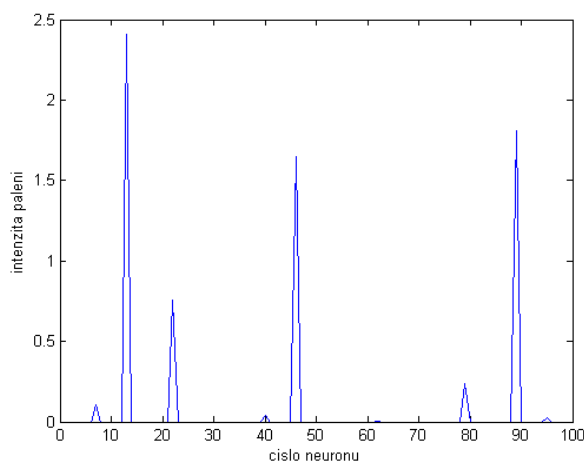
Na vstupní vektor \mathbf{x}_7 s doteky z vektorů $\mathbf{x}_2, \mathbf{x}_4, \mathbf{x}_5$ využiji při hledání nejsilněji pálících neuronů princip laterální inhibice. Intenzita pálení jednotlivých neuronů je na obr. 12.

Na obr. 12 je vidět, že nejvíce pálí neurony 1 (dotek v \mathbf{x}_5), 46 (dotek v \mathbf{x}_2) a 89 (dotek v \mathbf{x}_4). Nejvíce pálící neurony odpovídají vítězným neuronům při samotných dotecích v $\mathbf{x}_2, \mathbf{x}_4, \mathbf{x}_5$.

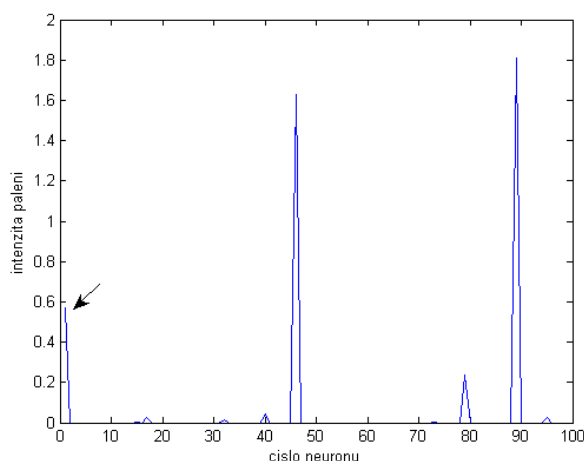
Interpretace a zhodnocení výsledků této série

V této sérii jsem chtěl ukázat, že používané grafické znázornění mapy je v souladu s pálením neuronů na doteky s aktivovanými taxely z grafického znázornění mapy (viz 5.2). Dále jsem chtěl ukázat, že funguje princip laterální inhibice v případě více současných doteků (viz 5.2.2).

Z výsledků 1. simulace je vidět, že v případě jednoho doteku s $dotykPst = 1$ ve



Obrázek 11. Znázornění intenzity pálení neuronů na vstupní vektor x_6 s více doteky při využití principu laterální inhibice



Obrázek 12. Znázornění intenzity pálení neuronů na vstupní vektor x_7 s více doteky při využití principu laterální inhibice

vstupním vektoru vyhrál vždy neuron, který se dal očekávat podle grafického znázornění mapy na obr. 10. V případě, že ve vstupním vektoru bylo více od sebe vzdálených doteků, byl využit princip laterální inhibice. Na vstupní vektor s více doteky páli silně více neuronů. Výrazně silněji pálicí neurony oproti ostatním odpovídaly každý jednomu doteku ve vstupním vektoru. V případě, že jeden z více současných doteků byl z rohu kůže a poměrně blízko něj byly další doteky, tak neuron příslušející tomuto taxelu vůbec nepálil. V případě, že okolo tohoto doteku v rohu kůže nebyly relativně blízko další doteky, tak jemu příslušející neuron už páli docela silně. To je způsobeno více vlivy. Za prvé rohový dotek aktivoval přibližně polovinu taxelů než ostatní vnitřní doteky. Další důvod je, že okraje kůže jsou slaběji reprezentované. To je způsobeno hraničním efektem (viz 3.4) a pak také tím, že při použitím způsobu generování doteků jsou krajní oblasti méně často stimulované než vnitřní oblasti. To vede k tomu, že blízký více pálicí neuron laterální inhibicí úplně utlumí neuron reprezentující okraj kůže.

6.2.3. 2. série simulací: poměr velikostí mapy a kůže

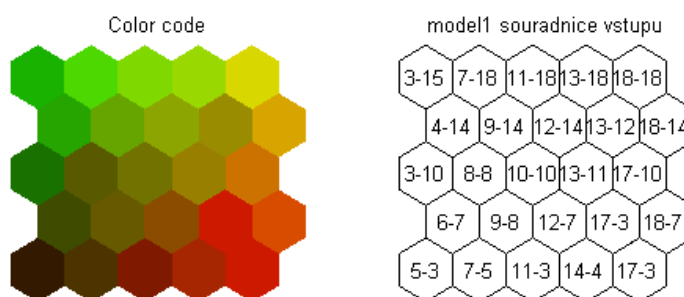
V této sérii budu zjišťovat, zda je neuronová síť schopna se učit při různých poměrech velikosti mapy (počtu neuronů) a velikosti kůže (počtu taxelů).

1. simulace

Mapa má označení 1703p8. Parametry simulace jsou:

```
m=20; n=20; o=5; p=5; %rozmery kuze(m,n)/mapy(o,p)
%jednodoteky
pocTrDot=40000; %pocet trenovacich doteku
dotykPst=1; %pravdepodobnost aktivace nestredoveho taxelu doteku
pocTrEp=6; %pocet trenovacich epoch-parametr trainlen
radiusIni=3; radiusFin=1; %radius init/final
RF=ones(o*p,m*n); %receptive fields
```

Grafické znázornění jedné z takto naučených map (1703p8o1) je na obr. 13.



SOM 17-Mar-2014

Obrázek 13. Simulace 1, vlevo jsou neurony obarveny podle pozice jim přiřazených taxelů, vpravo jsou u neuronů uvedeny souřadnice jim přiřazených taxelů

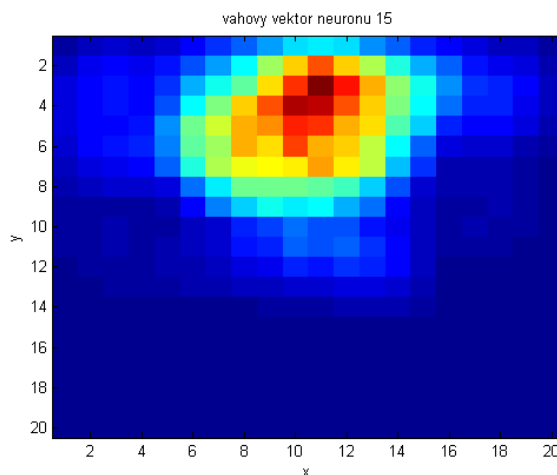
Z pěti provedených simulací ve všech případech mapa splňovala požadovanou vlastnost zachování topologie.

Na obr. 14 je váhový vektor neuronu 15 z mapy v obr. 13, zobrazený v rovině (funkce `vahyNeuronu`). Váhový vektor vypadá v pořádku.

2. simulace

Mapa má označení 1703p9. Parametry simulace jsou:

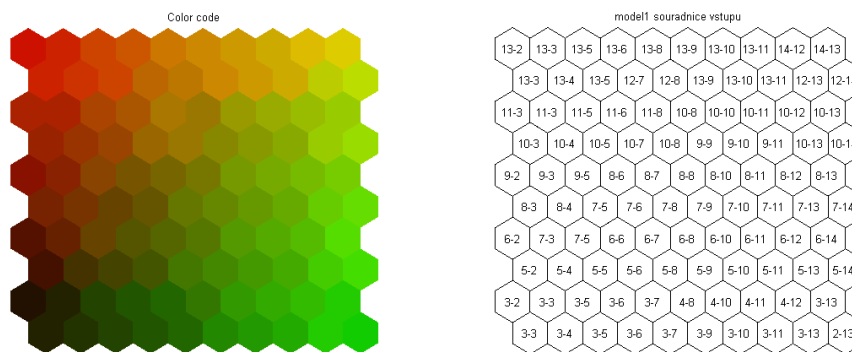
```
m=15; n=15; o=10; p=10; %rozmery kuze(m,n)/mapy(o,p)
```



Obrázek 14. Simulace 1, váhový vektor neuronu 15 zobrazený v rovině

```
%jednodoteky
pocTrDot=40000; %pocet trenovacich doteku
dotykPst=1; %pravdepodobnost aktivace nestredoveho taxelu doteku
pocTrEp=6; %pocet trenovacich epoch-parametr trainlen
radiusIni=3; radiusFin=1; %radius init/final
RF=ones(o*p,m*n); %receptive fields
```

Grafické znázornění jedné z takto naučených map (1703p9o2) je na obr. 15. Grafické



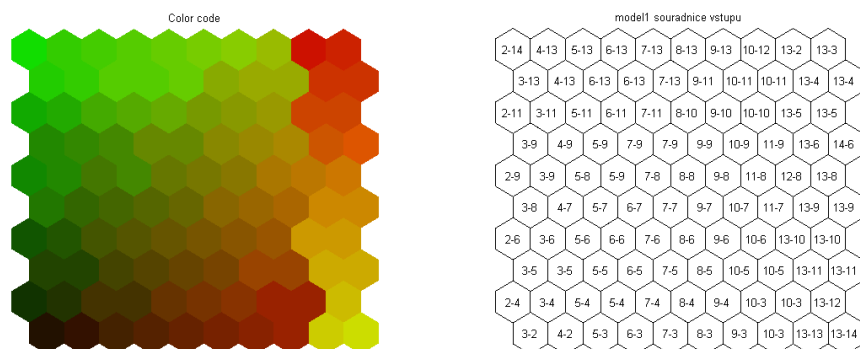
SOM 18-Mar-2014

Obrázek 15. Simulace 2, mapa 1, vlevo jsou neurony obarveny podle pozice jim přiřazených taxelů, vpravo jsou u neuronů uvedeny souřadnice jim přiřazených taxelů

znázornění další z takto naučených map (1703p9o5) je na obr. 16. Z pěti provedených simulací ve čtyřech případech mapa splňovala požadovanou vlastnost zachování topologie. Případ nesplnění vlastnosti zachování topologie je na obr. 16.

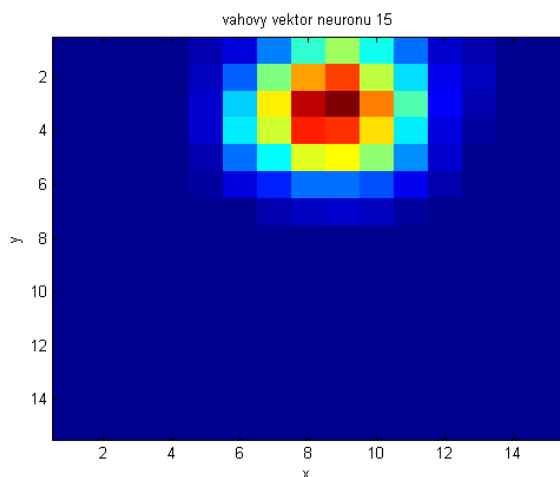
Na obr. 17 je váhový vektor neuronu 15 mapy z obr. 15 zobrazený v rovině. Váhový

6. Testování naučených neuronových sítí a interpretace výsledků



SOM 18-Mar-2014

Obrázek 16. Simulace 2, mapa 2, vlevo jsou neurony obarveny podle pozice jim přiřazených taxelů, vpravo jsou u neuronů uvedeny souřadnice jim přiřazených taxelů



Obrázek 17. Simulace 2, váhový vektor neuronu 15 zobrazený v rovině

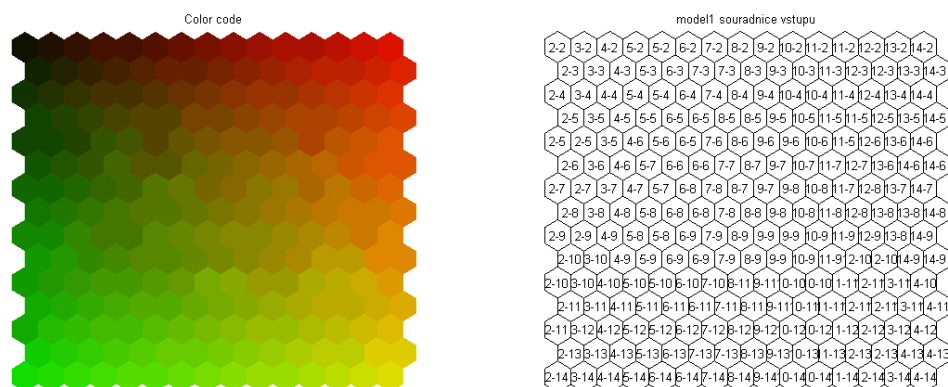
vektor vypadá v pořádku.

3. simulace

Mapa má označení 1803p1. Parametry simulace jsou:

```
m=15; n=15; o=15; p=15; %rozmery kuze(m,n)/mapy(o,p)
%jednodoteky
pocTrDot=40000; %pocet trenovacich doteku
dotykPst=1; %pravdepodobnost aktivace nestredoveho taxelu doteku
pocTrEp=6; %pocet trenovacich epoch-parametr trainlen
radiusIni=3; radiusFin=1; %radius init/final
RF=ones(o*p,m*n); %receptive fields
```

Grafické znázornění jedné z takto naučených map (1803p1o2) je na obr. 18.

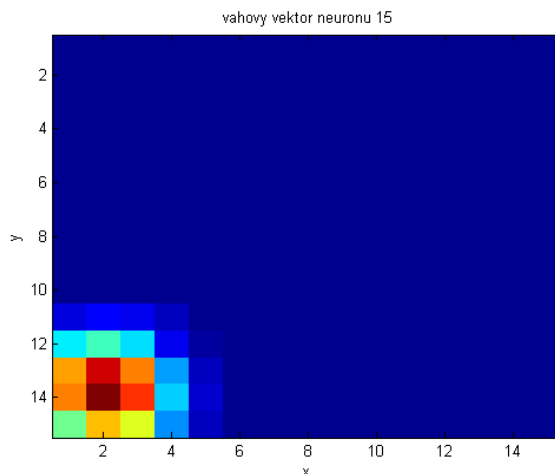


SOM 18-Mar-2014

Obrázek 18. Simulace 3, vlevo jsou neurony obarveny podle pozice jim přiřazených taxelů, vpravo jsou u neuronů uvedeny souřadnice jim přiřazených taxelů

Z pěti provedených simulací ve všech případech mapa splňovala požadovanou vlastnost zachování topologie.

Na obr. 19 je váhový vektor neuronu 15 mapy z obr. 18 zobrazený v rovině. Váhový



Obrázek 19. Simulace 3, váhový vektor neuronu 15 zobrazený v rovině

vektor vypadá v pořádku.

Interpretace a zhodnocení výsledků této série

Z výsledků této série lze vidět, že neuronová síť se dokázala naučit při různých poměrech velikosti mapy a kůže. V 1. simulaci byl celkový počet neuronů v mapě 25 a počet taxelů byl 400. To znamená, že na jeden neuron připadá v průměru 16 taxelů. V 2. simulaci

měla mapa 100 neuronů a kůže měla 225 taxelů. Na jeden neuron připadalo tedy v průměru 2.25 taxelu. Ve 3. simulaci měla mapa 225 neuronů a kůže měla 225 taxelů. To znamená, že na jeden neuron připadal v průměru 1 taxel.

Rozdílný průměrný počet taxelů připadající na jeden neuron byl vidět i v plošném znázornění váhového vektoru neuronu 15, uvedeného ve všech simulacích v této sérii. Pochopitelně počet taxelů, se kterými je neuron propojen, je zpravidla větší než uvedený průměrný počet. To je pochopitelné, protože části kůže, na které jsou citlivé jednotlivé neurony, se částečně překrývají.

Je zřejmé, že při potřebě většího rozlišení taxelů v jejich reprezentaci na mapě je potřeba také více neuronů v mapě. To nemusí být vždy výhodné, protože se zvyšujícím počtem neuronů v mapě roste časová náročnost učení.

6.2.4. 3. série simulací: dva současné doteky

V této sérii budu zkoumat, zda se mapa dobře naučí na vstupních vektorech se dvěma současnými doteky. Dále vyzkouším, zda míra pro hodnocení zachování topologie umožňuje rozlišit, zda mapa splňuje uvedenou vlastnost.

Testovací množina míry je vygenerovaná takto:

```
Mtest=createTouches(m,n,500,0.5);
```

To znamená, že obsahuje 500 vektorů a parametr *dotykPst* = 0.5. Lze ji nalézt na příloženém CD pod názvem *testMnoz05*, stejně jako funkci *createTouches*.

1. simulace

Mapa má označení 2403p1. Parametry simulace jsou:

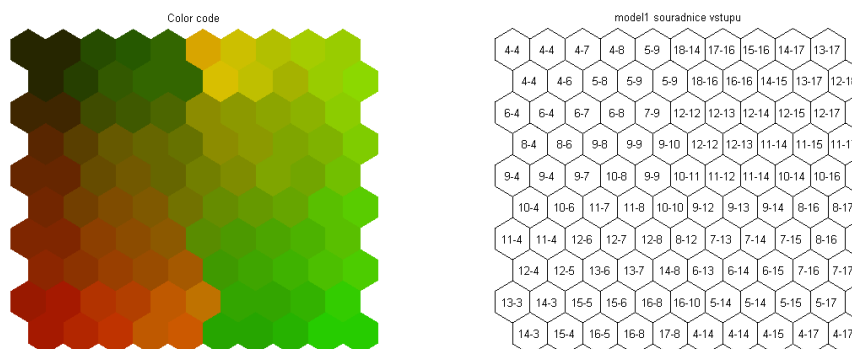
```
m=20; n=20; o=10; p=10; %rozmery kuze(m,n)/mapy(o,p)
%dvojdoteky
pocTrDot=20000; %pocet trenovacich doteku
dotykPst=1; %pravdepodobnost aktivace nestredoveho taxelu doteku
pocTrEp=6; %pocet trenovacich epoch-parametr trainlen
radiusIni=5; radiusFin=1; %radius init/final
RF=ones(o*p,m*n); RF(51:100,1:180)=0; %receptive fields
RF(1:50,221:400)=0;
```

Grafické znázornění jedné z takto naučených map (2403p1o1) je na obr. 20. Hodnoty míry pro hodnocení zachování topologie vstupní vrstvy ve výstupní vrstvě jsou shrnuty v Tab. 1.

Tabulka 1. Simulace 1, tabulka s hodnotami míry pro hodnocení zachování topologie vstupní vrstvy ve výstupní vrstvě

opakování	ukazatel mean	ukazatel std	správně
1	1.24	1.66	ne
2	0.90	0.76	ano
3	1.16	1.48	ne
4	1.29	2.00	ne
5	0.91	0.74	ano

Z pěti provedených simulací jen ve dvou případech mapa splňovala požadovanou vlastnost zachování topologie.



SOM 25-Mar-2014

Obrázek 20. Simulace 1, vlevo jsou neurony obarveny podle pozice jim přiřazených taxelů, vpravo jsou u neuronů uvedeny souřadnice jim přiřazených taxelů

2. simulace

Mapa má označení 2503p1. Parametry simulace jsou:

```
m=20; n=20; o=10; p=10; %rozmery kuze(m,n)/mapy(o,p)
%dvojdoteky
pocTrDot=20000; %pocet trenovacich doteku
dotykPst=1; %pravdepodobnost aktivace nestredoveho taxelu doteku
pocTrEp=12; %pocet trenovacich epoch-parametr trainlen
radiusIni=5; radiusFin=1; %radius init/final
RF=ones(o*p,m*n); RF(51:100,1:180)=0; %receptive fields
RF(1:50,221:400)=0;
```

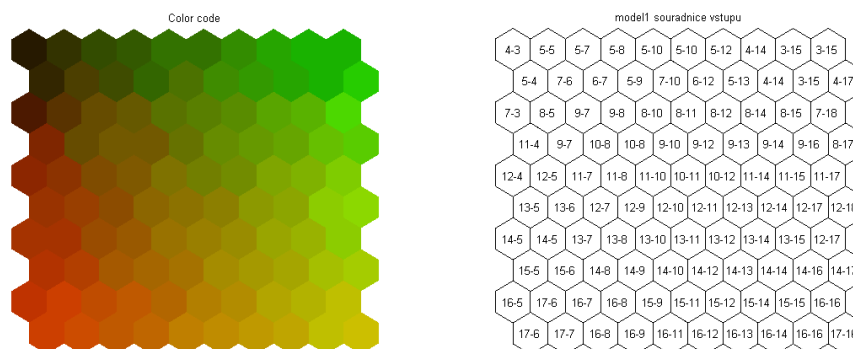
Grafické znázornění jedné z takto naučených map (2503p1o1) je na obr. 21. Hodnoty míry pro hodnocení zachování topologie vstupní vrstvy ve výstupní vrstvě jsou shrnuty v Tab. 2.

Tabulka 2. Simulace 2, tabulka s hodnotami míry pro hodnocení zachování topologie vstupní vrstvy ve výstupní vrstvě

opakování	ukazatel mean	ukazatel std	správně
1	0.93	0.73	ano
2	0.91	0.77	ano
3	0.91	0.73	ano
4	0.88	0.70	ano
5	0.91	0.70	ano

Z pěti provedených simulací ve všech případech splňovala mapa požadovanou vlastnost zachování topologie.

6. Testování naučených neuronových sítí a interpretace výsledků



SOM 25-Mar-2014

Obrázek 21. Simulace 2, vlevo jsou neurony obarveny podle pozice jim přiřazených taxelů, vpravo jsou u neuronů uvedeny souřadnice jim přiřazených taxelů

3. simulace

Mapa má označení 2603p1. Parametry simulace jsou:

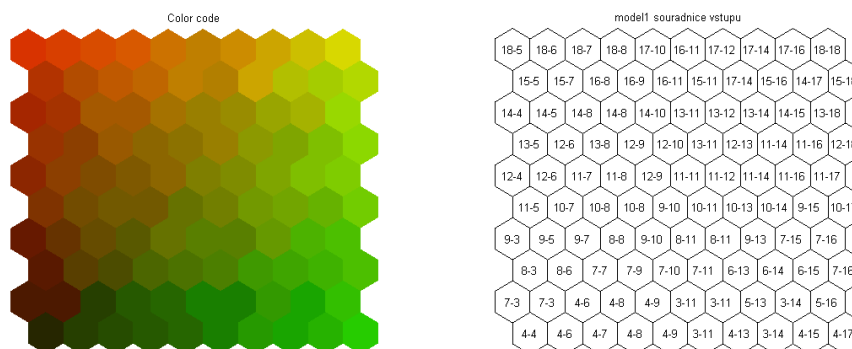
```
m=20; n=20; o=10; p=10; %rozmery kuze(m,n)/mapy(o,p)
%dvojdoteky
pocTrDot=20000; %pocet trenovacich doteku
dotykPst=0.2; %pravdepodobnost aktivace nestredoveho taxelu doteku
pocTrEp=12; %pocet trenovacich epoch-parametr trainlen
radiusIni=5; radiusFin=1; %radius init/final
RF=ones(o*p,m*n); RF(51:100,1:180)=0; %receptive fields
RF(1:50,221:400)=0;
```

Grafické znázornění jedné z takto naučených map (2603p1o1) je na obr. 22. Hodnoty míry pro hodnocení zachování topologie vstupní vrstvy ve výstupní vrstvě jsou shrnuty v Tab. 3.

Tabulka 3. Simulace 3, tabulka s hodnotami míry pro hodnocení zachování topologie vstupní vrstvy ve výstupní vrstvě

opakování	ukazatel mean	ukazatel std	správně
1	0.91	0.73	ano
2	0.88	0.72	ano
3	0.92	0.73	ano
4	1.18	1.63	ne
5	0.93	0.72	ano

Z pěti provedených simulací ve čtyřech případech mapa splňovala požadovanou vlastnost zachování topologie.



SOM 26-Mar-2014

Obrázek 22. Simulace 3, vlevo jsou neurony obarveny podle pozice jim přiřazených taxelů, vpravo jsou u neuronů uvedeny souřadnice jim přiřazených taxelů

4. simulace

Mapa má označení 2603p3. Parametry simulace jsou:

```

m=20; n=20; o=10; p=10; %rozmetry kuze(m,n)/mapy(o,p)
%dvojdoteky
pocTrDot=20000; %pocet trenovacich doteku
dotykPst=1; %pravdepodobnost aktivace nestredoveho taxelu doteku
pocTrEp=12; %pocet trenovacich epoch-parametr trainlen
radiusIni=5; radiusFin=1; %radius init/final
RF=ones(o*p,m*n); RF(51:100,1:200)=0; %receptive fields
RF(1:50,201:400)=0;

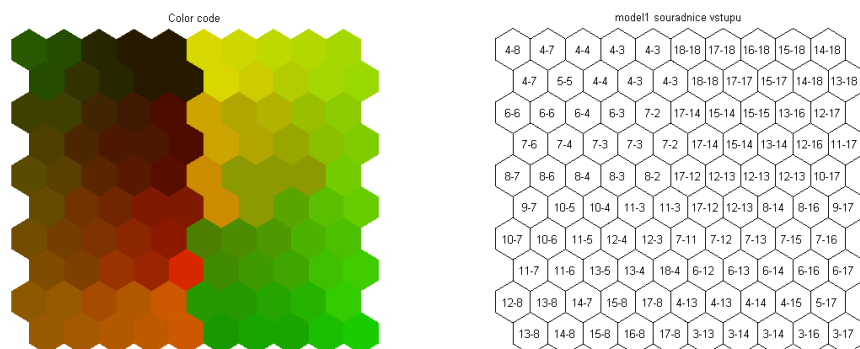
```

Grafické znázornění jedné z takto naučených map (2603p1o1) je na obr. 23. Hodnoty míry pro hodnocení zachování topologie vstupní vrstvy ve výstupní vrstvě jsou shrnuty v Tab. 4.

Tabulka 4. Simulace 4, tabulka s hodnotami míry pro hodnocení zachování topologie vstupní vrstvy ve výstupní vrstvě

opakování	ukazatel mean	ukazatel std	správně
1	1.23	1.70	ne
2	1.11	1.39	ne
3	1.09	1.23	ne
4	1.13	1.50	ne
5	0.90	0.76	ano

Z pěti provedených simulací jen v jednom případě splňovala mapa požadovanou vlastnost zachování topologie.



SOM 26-Mar-2014

Obrázek 23. Simulace 4, vlevo jsou neurony obarveny podle pozice jim přiřazených taxelů, vpravo jsou u neuronů uvedeny souřadnice jim přiřazených taxelů

Interpretace a zhodnocení výsledků této série

V této sérii byly v trénovací množině pouze vstupní vektory se dvěma současnými doteky. Hlavním cílem bylo ukázat, že se mapa dokáže z takové trénovací množiny naučit.

Z 1. simulace a 2. simulace je vidět, že počet trénovacích epoch je pro úspěch učení mapy zásadní. Dvojnásobné zvětšení trénovacích epoch zvýšilo úspěšnost ze dvou dobře naučených map z pěti, na pět dobře naučených map z pěti.

Ve 3. simulaci je prezentováno, že se mapa dokázala naučit i na poměrně malých dotecích. V jednom doteku bylo průměrně aktivováno jen 2.6 taxelu ($dotykPst = 0.2$). Jinak se tato simulace nelišila od 2. simulace.

Ve 4. simulaci je vidět, že u dvou současných doteků ve vstupních vektorech v trénovací množině hraje překryv receptivních polí důležitou roli. V této simulaci se receptivní pole půlek map nepřekrývaly (jinak je stejná jako 2. simulace). Při jednom současném doteku to moc nevadilo, jak se lze přesvědčit v 1. sérii simulací (viz 6.2.2).

Dále si lze všimnout, že použitá míra zachování topologie vstupní vrstvy ve výstupní vrstvě umožňovala velmi dobře rozlišit, zda je mapa správně naučená nebo ne. Největší hodnota míry správně naučené mapy byla 0.93 a nejmenší hodnota míry nesprávně naučené mapy byla 1.09. Je tedy vidět, že vzhledem k tomu, v jakých hodnotách se hodnota míry pohybuje, je mezi oběma případy velká mezera, a proto je možné s velkou jistotou podle hodnoty míry určit, zda je mapa naučená správně nebo ne.

6.2.5. 4. série simulací: vliv četnosti stimulace taxelu

V této sérii budu výrazně více stimulovat určitou oblast kůže a budu chtít zjistit, jak se to projeví v naučené mapě.

1. simulace

Mapa má označení 0205p1. Parametry simulace jsou:

```

m=12; n=12; o=7; p=7; %rozmetry kuze(m,n)/mapy(o,p)
dotykPst=1;
pocTrEp=6; %pocet trenovacich epoch-parametr trainlen
radiusIni=5; radiusFin=1; %radius init/final
RF=ones(o*p,m*n); %receptive fields

```

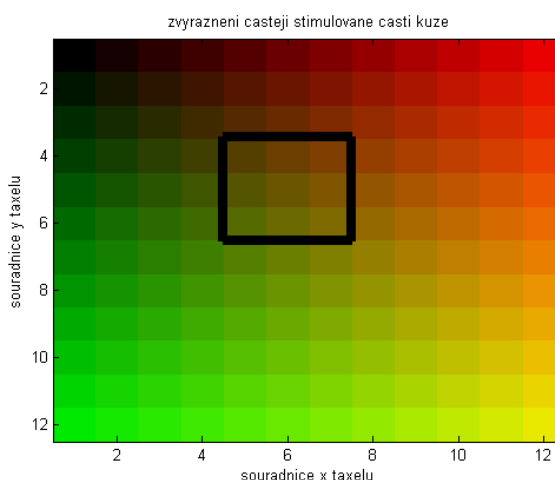
Vytvoření trénovací množiny je trochu složitější. V trénovací množině je 20000 náhodně vytvořených vektorů s jednodoteky rovnoměrně rozloženými po celé kůži. Dále jsou navíc v trénovací množině vektory s doteky o 9 taxelech ($dotykPst = 1$) s těmito středy:

```

(5,4), (5,5), (5,6)
(6,4), (6,5), (6,6)
(7,4), (7,5), (7,6)

```

Každý z těchto doteků je v trénovací množině 2000 krát. To znamená, že celkový počet trénovacích vektorů je $20000 + 9 * 2000 = 38000$. To znamená, že stimulace oblasti kůže uvedených taxelů je velmi intenzivní. Stimulovaná oblast kůže (středů zmíněných doteků) je zvýrazněna černým obdélníkem na obr. 24.

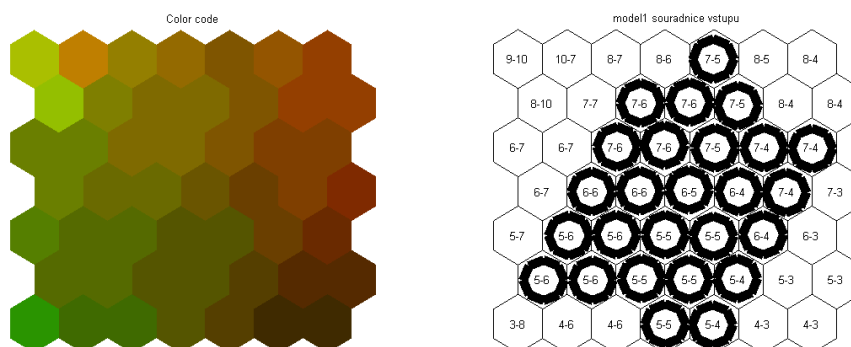


Obrázek 24. Kůže se zvýrazněním (černý obdélník) častěji stimulovaných taxelů

Grafické znázornění jedné z takto naučených map (0205p1o1) je na obr. 25. V pravé části obr. 25 jsou znázorněny neurony, které mají přiřazen některý z devíti taxelů z častěji stimulované oblasti kůže.

Interpretace a zhodnocení výsledků této série

V této sérii šlo o to ukázat, že neuronová síť splňuje vlastnost uvedenou v 2.3.2. To znamená, že častěji stimulované oblasti kůže jsou reprezentovány více neurony. To se podařilo jednoznačně ukázat. Při pohledu na zastoupení častěji stimulované oblasti kůže v černém čtverečku z obr. 24 je na mapě na obr. 25 (vpravo - černá kolečka) jednoznačně vidět, že častěji stimulovaná oblast kůže zabírá většinu neuronů v mapě.



SOM 02-May-2014

Obrázek 25. Simulace 1, v pravé části jsou v mapě znázorněny neurony (černá kolečka) s taxely z častěji stimulované oblasti kůže z obr. 24

6.2.6. 5. série simulací: šest současných doteků

V této sérii budu zkoumat, zda se mapa dobře naučí na vstupních vektorech se šesti současnými doteky. Dále předvedu vliv velikosti doteků (počtu aktivovaných taxelů) a vliv velikosti trénovací množiny na učení.

1. simulace

Mapa má označení 3003p2. Parametry simulace jsou:

```
m=20; n=20; o=10; p=10; %rozmery kuze(m,n)/mapy(o,p)
%sestidoteky
pocTrDot=20000; %pocet trenovacich doteku
dotykPst=0.5; %pravdepodobnost aktivace nestredoveho taxelu doteku
pocTrEp=24; %pocet trenovacich epoch-parametr trainlen
radiusIni=5; radiusFin=1; %radius init/final
RF=ones(o*p,m*n); RF(51:100,1:180)=0; %receptive fields
RF(1:50,221:400)=0;
```

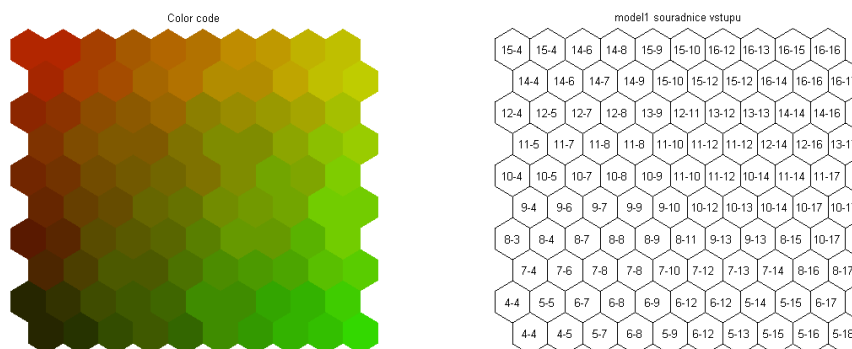
Grafické znázornění jedné z takto naučených map (3003p2o1) je na obr. 26.

Z pěti provedených simulací ve čtyřech případech mapa splňovala požadovanou vlastnost zachování topologie.

2. simulace

Mapa má označení 3103p1. Parametry simulace jsou:

```
m=20; n=20; o=10; p=10; %rozmery kuze(m,n)/mapy(o,p)
%sestidoteky
pocTrDot=20000; %pocet trenovacich doteku
dotykPst=0.2; %pravdepodobnost aktivace nestredoveho taxelu doteku
```

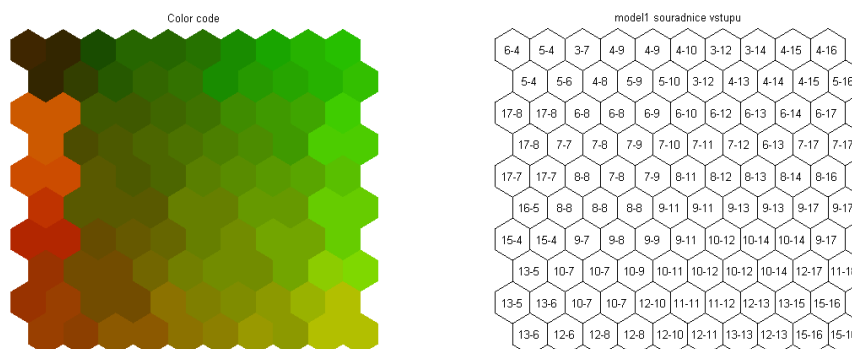


SOM 30-Mar-2014

Obrázek 26. Simulace 1, vlevo jsou neurony obarveny podle pozice jim přiřazených taxelů, vpravo jsou u neuronů uvedeny souřadnice jim přiřazených taxelů

```
pocTrEp=24; %pocet trenovacich epoch-parametr trainlen
radiusIni=5; radiusFin=1; %radius init/final
RF=ones(o*p,m*n); RF(51:100,1:180)=0; %receptive fields
RF(1:50,221:400)=0;
```

Grafické znázornění jedné z takto naučených map (3103p1o1) je na obr. 27.



SOM 31-Mar-2014

Obrázek 27. Simulace 2, vlevo jsou neurony obarveny podle pozice jim přiřazených taxelů, vpravo jsou u neuronů uvedeny souřadnice jim přiřazených taxelů

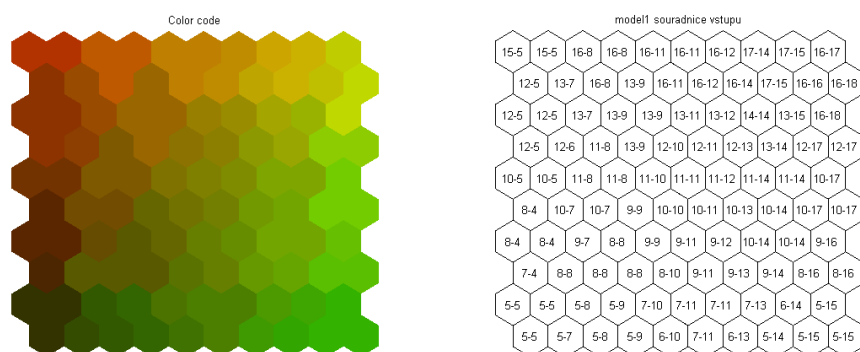
Z pěti provedených simulací v žádném případě mapa nesplňovala požadovanou vlastnost zachování topologie.

3. simulace

Mapa má označení 0604p1. Parametry simulace jsou:

```
m=20; n=20; o=10; p=10; %rozmery kuze(m,n)/mapy(o,p)
%sestidoteky
pocTrDot=40000; %pocet trenovacich doteku
dotykPst=0.2; %pravdepodobnost aktivace nestredoveho taxelu doteku
pocTrEp=24; %pocet trenovacich epoch-parametr trainlen
radiusIni=5; radiusFin=1; %radius init/final
RF=ones(o*p,m*n); RF(51:100,1:180)=0; %receptive fields
RF(1:50,221:400)=0;
```

Grafické znázornění jedné z takto naučených map (0604p1o1) je na obr. 28.



SOM 06-Apr-2014

Obrázek 28. Simulace 3, vlevo jsou neurony obarveny podle pozice jim přiřazených taxelů, vpravo jsou u neuronů uvedeny souřadnice jim přiřazených taxelů

Z pěti provedených simulací ve čtyřech případech mapa splňovala požadovanou vlastnost zachování topologie.

Interpretace a zhodnocení výsledků této série

V této sérii jsem chtěl hlavně ukázat, že se neuronová síť zvládne naučit i na vstupních vektorech se šesti současnými doteky.

V 1. simulaci v jednom doteku bylo průměrně aktivováno 5 taxelů ($dotykPst = 0.5$). Při daných uvedených parametrech se mapa naučila správně ve čtyřech případech z pěti. Ve 2. simulaci byly parametry obdobné, jen se zmenšily doteky. V doteku pak bylo v průměru aktivováno 2.6 taxelů ($dotykPst = 0.2$). Z pěti provedených simulací se mapa správně nenaučila ani v jediném případě. Pokud se dvojnásobně zvětšila velikost trénovací množiny (a tím i doba učení), tak se mapa naučila správně ve čtyřech případech z pěti. To je prezentováno ve 3. simulaci.

6.2.7. 6. série simulací: nastavení receptivních polí, velká kůže

V této sérii předvedu učení mapy na větší kůži. Dále se budu zabývat vhodným nastavením receptivních polí a vlivem počtu neuronů v mapě na učení.

Testovací množina míry je vygenerovaná takto:

```
MtestB=createTouches(m,n,500,0.5);
```

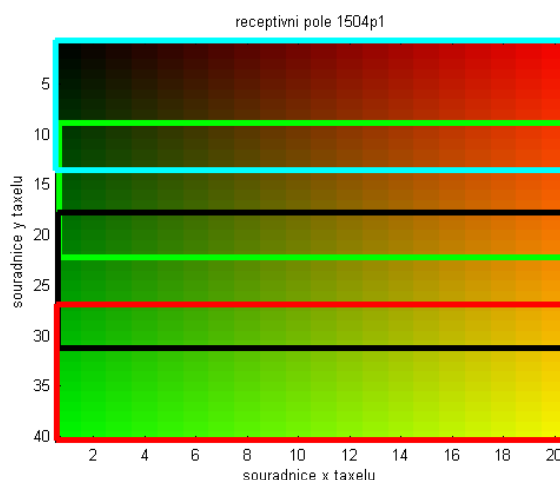
To znamená, že obsahuje 500 vektorů a parametr *dotykPst* = 0.5. Lze ji nalézt na příloženém CD pod názvem *testMnoz05BigSkin*, stejně jako funkci *createTouches*.

1. simulace

Mapa má označení 1504p1. Parametry simulace jsou:

```
m=40; n=20; o=10; p=10; %rozmetry kuze(m,n)/mapy(o,p)
%trojdoteky
pocTrDot=100000; %pocet trenovacich doteku
dotykPst=1; %pravdepodobnost aktivace nestredoveho taxelu doteku
pocTrEp=24; %pocet trenovacich epoch-parametr trainlen
radiusIni=5; radiusFin=1; %radius init/final
```

Protože nastavení receptivních polí je trochu složitější (viz CD), na obr. 29 je vyznačena kůže včetně nastavení receptivních polí.



Obrázek 29. Kůže s vyznačenými receptivními poli ze simulace 1

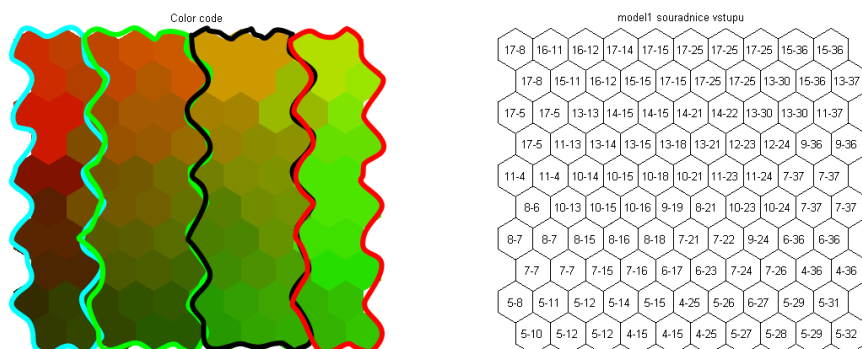
Grafické znázornění jedné z takto naučených map (1504p1o1) včetně vyznačení příslušnosti neuronů k receptivním polím je na obr. 30.

Z pěti provedených simulací v žádném případě mapa nespĺňovala požadovanou vlastnost zachování topologie.

2. simulace

Mapa má označení 1904p2. Parametry simulace jsou:

6. Testování naučených neuronových sítí a interpretace výsledků

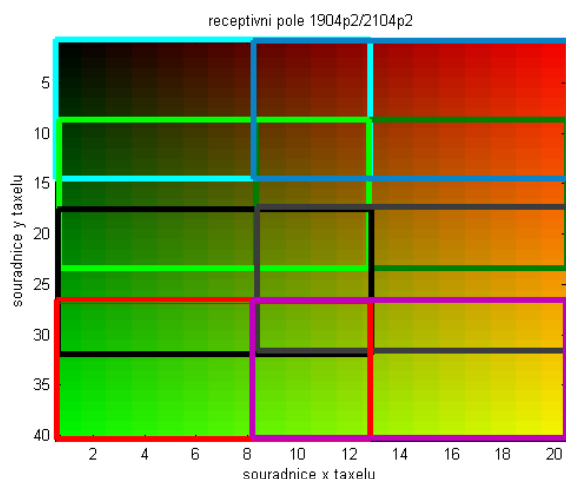


SOM 15-Apr-2014

Obrázek 30. Simulace 1, vlevo jsou neurony obarveny podle pozice jim přiřazených taxelů, vpravo jsou u neuronů uvedeny souřadnice jim přiřazených taxelů. Dále na obrázku vlevo je vyznačena příslušnost neuronů k receptivním polím z obr. 29

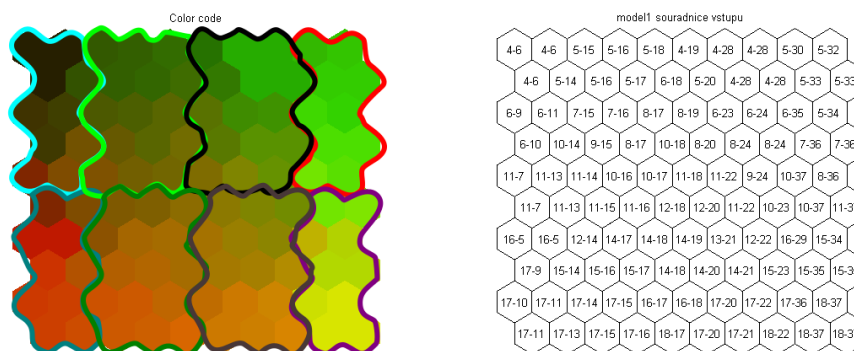
```
m=40; n=20; o=10; p=10; %rozmery kuze(m,n)/mapy(o,p)
%trojdoteky
pocTrDot=100000; %pocet trenovacich doteku
dotykPst=1; %pravdepodobnost aktivace nestredoveho taxelu doteku
pocTrEp=24; %pocet trenovacich epoch-parametr trainlen
radiusIni=5; radiusFin=1; %radius init/final
```

Protože nastavení receptivních polí je trochu složitější (viz příložené CD) na obr. 31 je vyznačena kůže včetně nastavení receptivních polí.



Obrázek 31. Kůže s vyznačenými receptivními poli ze simulace 2 a simulace 3

Grafické znázornění jedné z takto naučených map (1904p2o1) včetně vyznačení příslušnosti k receptivním polím je na obr. 32.



SOM 19-Apr-2014

Obrázek 32. Simulace 2, vlevo jsou neurony obarveny podle pozice jim přiřazených taxelů, vpravo jsou u neuronů uvedeny souřadnice jim přiřazených taxelů. Dále na obrázku vlevo je vyznačena příslušnost neuronů k receptivním polím z obr. 31

Z pěti provedených simulací v žádném případě mapa nesplňovala požadovanou vlastnost zachování topologie.

3. simulace

Mapa má označení 2104p2. Parametry simulace jsou:

```
m=40; n=20; o=10; p=20; %rozmetry kuze(m,n)/mapy(o,p)
%trojdoteky
pocTrDot=100000; %pocet trenovacich doteku
dotykPst=1; %pravdepodobnost aktivace nestredoveho taxelu doteku
pocTrEp=24; %pocet trenovacich epoch-parametr trainlen
radiusIni=5; radiusFin=1; %radius init/final
```

Protože nastavení receptivních polí je trochu složitější (viz příložené CD) na obr. 31 je vyznačena kůže včetně nastavení receptivních polí (stejně jako v simulaci 2).

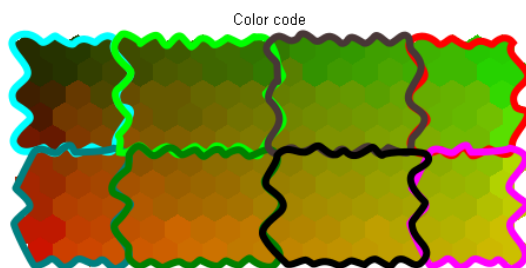
Grafické znázornění jedné z takto naučených map (2104p2o2) včetně vyznačení příslušnosti k receptivním polím je na obr. 33.

Z pěti provedených simulací ve všech případech mapa splňovala požadovanou vlastnost zachování topologie.

Interpretace a zhodnocení výsledků této série

V této sérii simulací šlo o to hlavně ukázat, že je mapa schopná se naučit i na kůži (40 * 20 taxelů), která je přibližně velká jako polovina kůže (což odpovídá sensorické mapě jedné hemisféry) robota iCub.

Z 1. a 2. simulace lze usuzovat, že vhodné nastavení receptivních polí zlepšuje učení. Konkrétně ve 2. simulaci, kde byly menší receptivní pole s dostatečnými překryvy, bylo výrazně méně diskontinuit v reprezentaci ve středu mapy. Diskontinuity byly hlavně na krajích mapy. Většinou se jednalo pouze o "nespojité" přechody ve správném směru.



SOM 21-Apr-2014

Obrázek 33. Simulace 3, neurony jsou obarveny podle pozice jim přiřazených taxelů. Dále je vyznačena příslušnost neuronů k receptivním polím z obr. 31

Ve 3. simulaci se oproti 2. simulaci dvojnásobně zvýšil počet neuronů na mapě. Tomu se uzpůsobilo přiřazení receptivních polí neuronům (viz obr. 33). To vedlo k výraznému zlepšení učení map. Z toho se dá usuzovat, že volba počtu neuronů mapy má významný vliv na kvalitu naučení mapy.

6.3. Učení s doteky z robota iCub

Požadovanou vlastností vytvořeného modelu neuronové sítě je použitelnost pro biologicky inspirovanou vnitřní reprezentaci taktilních sensorů robota iCub. Podle výsledků simulací s umělými doteky má pro tento účel neuronová síť dobré předpoklady. V rámci této práce se podařilo provést i simulace s trénovací množinou vytvořenou na robotovi iCub. Časové možnosti této práce umožnily provést pouze poměrně omezený počet těchto simulací. Trénovací množina obsahovala převážně jednodoteky.

Z uvedených důvodů uvedu pouze několik simulací, které mají ilustrovat základní vlastnosti map naučených na reálných dotecích z robota iCub. Podoba kůže robota iCub je v některých ohledech významně odlišná od podoby kůže pro generování umělých doteků (méně souvislá, tvarově složitější). Proto se v naučených mapách objevily některé nové jevy, které nebylo možné v rozsahu této práce řádně prozkoumat.

Pro dobré porozumění struktuře kůže robota iCub je velmi vhodné nejdřív přečíst kapitolu C (příloha).

6.3.1. Popis trénovacích dat

Tvorba trénovacích dat probíhala dotýkáním se kůže robota iCub. Při tvorbě trénovacích dat byla kůže stimulována palcem ruky. Zpravidla byla v daný časový okamžik stimulována pouze jedna oblast kůže. Palec se při dotýkání pohyboval po kůži obdobně, jako kdyby se snažil po povrchu kůže rozetřít barvu. To znamená, že byl většinu času v kontaktu s kůží. Zároveň byla snaha, aby oblasti kůže byly stimulovány rovnoměrně. To se ukázalo jako obtížné realizovat, a proto jsou v trénovací množině taxely, které mají oproti ostatním taxelům výrazně více aktivací. To se pak projeví ve větším zastoupení neuronů reprezentujících tyto taxely v mapě.

Kůže robota iCub není souvislá. To znamená, že některé okraje sousedních částí kůže na sebe nenavazují (např. předloktí a nadloktí je odděleno kloubem, který není pokryt kůží). Proto běžným dotýkáním nedojde ke koaktivacím sousedních krajů těchto částí kůže. To vede k tomu, že reprezentace těchto částí v mapě na sebe nemůžou plynule navazovat. Tento problém se řešil tak, že se vytvořily doteky, kdy byly stimulovány sousední okraje nenavazujících oblastí kůže. Tímto způsobem mapa "získá dojem", že na sebe obě části kůže navazují.

Stimulovány byly části kůže: pravá dlaň, pravé předloktí, pravé nadloktí a přední část trupu. To jsou všechny části kůže přibližně jedné poloviny těla robota iCub (přední část trupu pokrývá obě poloviny těla).

Všechny použité trénovací množiny lze nalézt na přiloženém CD. K zobrazení průběhu stimulace kůže z trénovací množiny lze použít funkci `showTouches`.

6.3.2. Přehled simulací

Pro lepší orientaci v provedených simulacích uvedu přehled simulací s jejich krátkým popisem.

1. sim. V této simulaci se mapa tvoří pouze pro dlaň. Malý počet neuronů a taxelů umožní čtenáři udělat si dobrou představu o naučené mapě.
2. sim. V této simulaci se mapa tvoří pro dlaň a předloktí. V simulaci bude ilustrován problém vzniku diskontinuit při více částech kůže složitějšího geometrického tvaru.
3. sim. V této simulaci se mapa tvoří pro celou přední část trupu. Kvůli podobnosti tvaru této kůže s obdélníkovým tvarem kůže, ze simulací s umělými doteky, bude ukázáno, že i na reálných dotecích se mapa dokáže naučit obdobně hezky (dobře odráží topologii kůže) jako při simulacích s umělými doteky.
4. sim. V této simulaci se mapa tvoří pro dlaň pravé ruky, pravé předloktí, nadloktí a celou přední část trupu. Tedy pro polovinu celé kůže robota iCub (až na kůži přední části trupu, která pokrývá obě poloviny těla). Na mapu lze pak nahlížet jako na senzorickou mapu homuncula odpovídající jedné mozkové hemisféře.

6.3.3. 1. simulace: dlaň

V této simulaci se mapa tvoří pouze pro dlaň. Malý počet neuronů a taxelů umožní čtenáři udělat si dobrou představu o naučené mapě.

Mapa má označení 1005p1. Parametry simulace jsou:

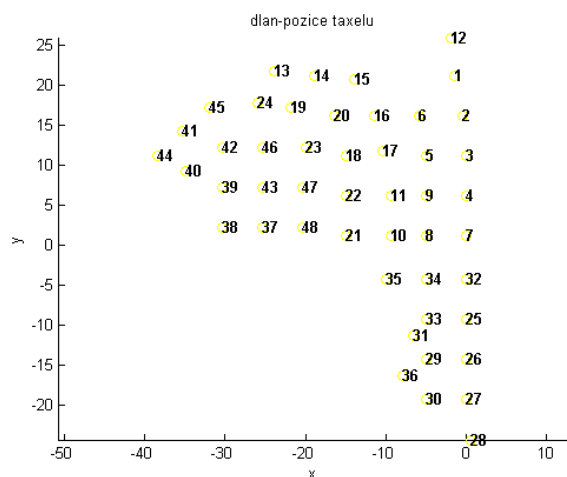
```
o=4; p=8; %rozmary mapy
pocTrEp=10; %pocet trenovacich epoch-parametr trainlen
radiusIni=5; radiusFin=1; %radius init/final
RF=ones(o*p,48);
Mtren(1:3000, :);
```

Trénovací množina `Mtren` je uložena v souboru `MtrenPalmSingle.mat`. Pozice taxelů z této simulace lze nalézt v souboru `taxPosPalmSingle.mat`. Oba soubory lze nalézt na přiloženém CD. Provedl jsem simulace (1005p2), kde se využila celá `Mtren`, ale protože výsledky vypadaly velmi podobně nebudu je zde uvádět.

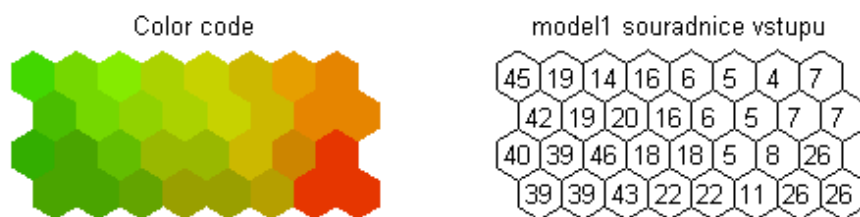
Kůže s očíslovanými taxely je znázorněna na obr. 34. K vykreslení byla použita funkce `plotTaxels`.

Grafické znázornění jedné z takto naučených map (1005p1o1) je na obr. 35. K vykreslení byla použita funkce `seeMapF3`.

6. Testování naučených neuronových sítí a interpretace výsledků



Obrázek 34. Pozice taxelů kůže na dlani

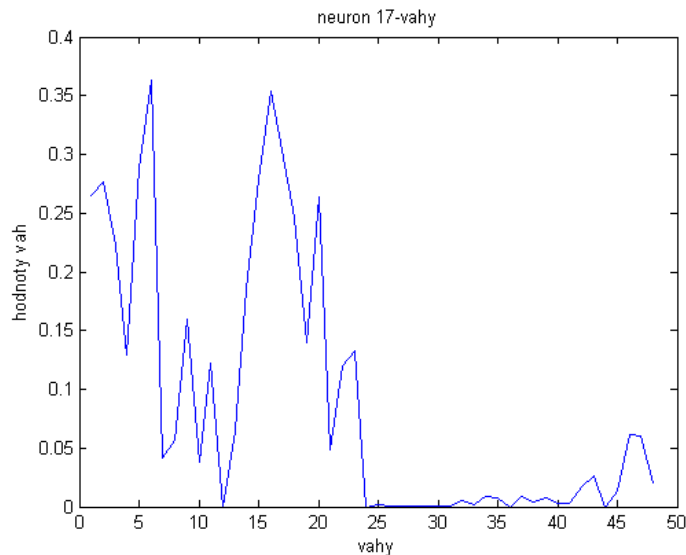


SOM 10-May-2014

Obrázek 35. Simulace 1, vlevo jsou neurony obarveny podle pozice jim přiřazených taxelů, vpravo jsou u neuronů uvedena čísla jim přiřazených taxelů

Váhový vektor neuronu 17 je znázorněn na obr. 36.

Z pěti provedených simulací ve všech případech mapa splňovala požadovanou vlastnost zachování topologie.



Obrázek 36. Simulace 1, váhový vektor neuronu 17

Interpretace a zhodnocení výsledků této simulace

Cílem této simulace bylo ukázat malou naučenou mapu na nejmenší části robotické kůže, na dlani. To umožní čtenáři přehledně se podívat, jak je mapa naučená.

V této simulaci se mapa naučila hezky jak se lze přesvědčit z obr. 35. Čtenář se může přesvědčit, že sousední neurony reprezentují blízké taxely z pravé části obr. 35 a pozic taxelů na obr. 34. Na znázorněném vektoru vah neuronu 17 (obr. 36) se lze přesvědčit, že neuron reprezentuje skutečně taxely, které jsou blízko sebe.

6.3.4. 2. simulace: dlaň a předloktí

V této simulaci se mapa tvoří pro dlaň a předloktí. V simulaci bude ilustrován problém vzniku diskontinuit při více částech kůže složitějšího geometrického tvaru.

Mapa má označení 1105p2. Parametry simulace jsou:

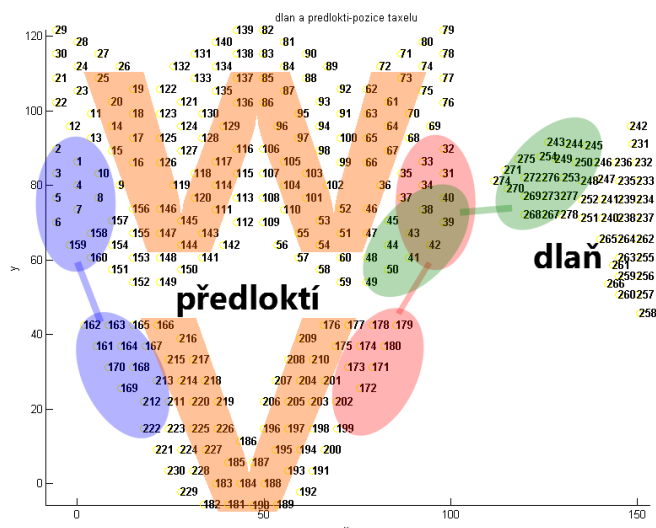
```
o=10; p=15; %rozmery mapy
pocTrEp=4; %pocet trenovacich epoch-parametr trainlen
radiusIni=5; radiusFin=1; %radius init/final
RF=zeros(o*p,278); RF(1:30,[231:278,41:44,48:50,31:34,38:40])=1;
RF(31:150,[1:230,243:245,249,254,267:276])=1; %receptive fields
Mtr=[MtrPomForearm;MtrPomPalm(1:4000,:);MtrForearmPalmSpoj(1500:2000,:);
```

Aby bylo možné vytvořit uvedenou trénovací množinu Mtr, je potřeba nejdříve načíst soubor `trenDataPosTax.mat`, který lze nalézt na příloženém CD. Uvedený soubor obsahuje také pozice použitých taxelů.

Kůže s očíslovanými taxely je znázorněna na obr. 37. Na obrázku je také znázorněno jak na sebe pláty kůže navazují (taxely uzavřené v elipsách stejné barvy na sebe navazují). Pláty kůže předloktí jsou navíc označeny písmeny, aby bylo možné později v mapě označit reprezentaci těchto dvou plátů kůže.

Grafické znázornění jedné z takto naučených map (1105p2o3) je na obr. 38. Popis lze nalézt u uvedeného obrázku.

6. Testování naučených neuronových sítí a interpretace výsledků



Obrázek 37. Pozice taxelů kůže na dlaní a předloktí. Uvnitř elips stejných barev jsou vyznačeny přibližně taxely, které jsou po navlečení kůže na iCuba blízko sebe (resp. koaktivacemi sousedních okrajů částí kůže to emulujeme).

Všechny naučené mapy v této simulaci měly několik diskontinuit. Vzhledem k tomu, že pravděpodobně tvar částí kůže neumožňuje reprezentaci v obdélníkové mapě bez diskontinuit, nelze jednoznačně tvrdit, že by se mapy naučily špatně.

Interpretace a zhodnocení výsledků této simulace

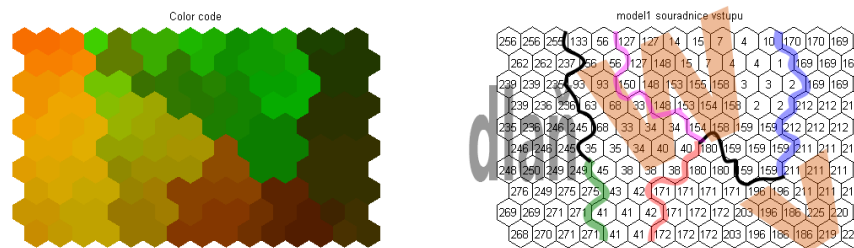
V této simulaci jsem chtěl hlavně ilustrovat problém vzniku diskontinuit při více částech kůže složitějšího geometrického tvaru. Na úlohu se dá pohlížet jako na úlohu z topologie, kdy máme několik geometrických objektů různých tvarů (v této simulaci pláty kůže o přibližném tvaru písmen V, W a obráceném P (dlaň)), které na sebe musí v určitých místech co nejvíce navazovat (tak jak na sebe navazují pláty kůže nasazené na iCubovi, resp. to koaktivacemi sousedních okrajů emulujeme). Navíc se ještě musí vejít všechny do obdélníku dané velikosti (reprezentace v mapě obdélníkového tvaru). Aby se to dalo realizovat, je nutné geometrické objekty (tvary) určitým způsobem zdeformovat. Jsem přesvědčen, že nelze zdeformovat geometrické objekty tak, aby byly splněny uvedené vlastnosti a nevznikly navíc nějaké diskontinuity (nebyly blízko sebe části geometrických objektů, které původně blízko sebe nebyly).

Je možné, že by šlo přednastavit receptivní pole tak, aby se počet diskontinuit zmenšil (v případě pouze tří plátů kůže možná i úplně eliminoval).

Části kůže dlaně a předloktí na iCubovi na sebe ve skutečnosti nenavazují. Aby na mapě reprezentace obou částí kůže navazovaly, bylo nutné udělat doteky s koaktivacemi blízkých okrajů dlaně a předloktí. V naučené mapě je vidět, že taxely okrajů předloktí a dlaně, které byly koaktivovány (viz obr. 37), na sebe v reprezentaci na mapě skutečně navazují.

6.3.5. 3. simulace: přední část trupu

V této simulaci se mapa tvoří pro celou přední část trupu. Kvůli podobnosti tvaru této kůže s obdélníkovým tvarem kůže, ze simulací s umělými doteky, bude ukázáno, že i na



SOM 11-May-2014

Obrázek 38. Simulace 2, vlevo jsou neurony obarveny podle pozice jim přiřazených taxelů, vpravo jsou u neuronů uvedena čísla jim přiřazených taxelů. Dále jsou barevnými čarami vyznačeny přechody mezi reprezentací plátů kůže. Barvy přechodů mezi reprezentací plátů kůže odpovídají barvám elips, které označují přechody, z obr. 37. Černé čáry znázorňují hranice mezi pláty kůže, kdy přechod není plynulý (tzn. reprezentované taxely na robotovi nejsou blízko sebe). Čára fialové barvy uvnitř reprezentace plátu předloktí označeném písmenem W znázorňuje diskontinuitu uvnitř reprezentace tohoto plátu.

reálných dotecích se mapa dokáže naučit obdobně hezky (dobře odráží topologii kůže) jako při simulacích s umělými doteky.

Mapa má označení 1105p3. Parametry simulace jsou:

```
o=10; p=15; %rozmary mapy
pocTrEp=20; %pocet trenovacich epoch-parametr trainlen
radiusIni=5; radiusFin=1; %radius init/final
RF=ones(o*p,440);
```

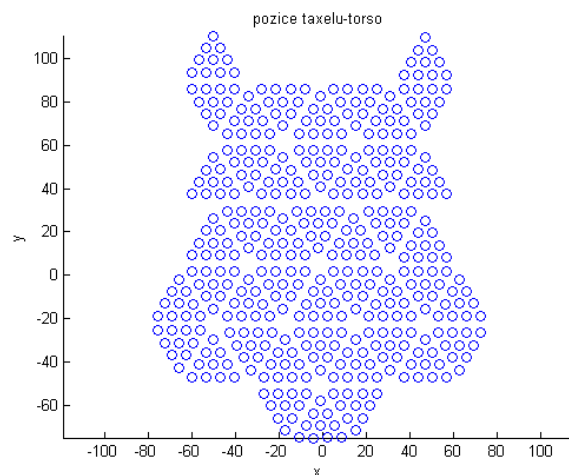
Trénovací množina je uložena v souboru `MtrBTorsoSingle.mat`. Pozice taxelů z této simulace lze nalézt v souboru `taxPosTorsoSingle.mat`. Oba soubory lze nalézt na příloženém CD.

Protože tato část kůže je poměrně velká, nebudu zde uvádět obrázek taxelů s jejich očíslováním. Tento obrázek je možné nalézt na příloženém CD pod názvem `torsoTaxPos.fig`. Na obr. 39 jsou vykresleny pozice taxelů bez jejich popisků. Tento obrázek slouží pouze pro rychlou představu o tvaru kůže přední části trupu.

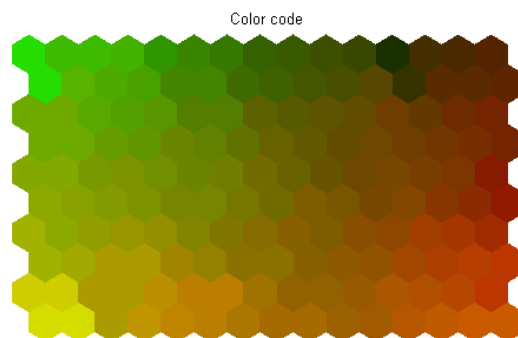
Grafické znázornění jedné z takto naučených map (1105p3o1) je na obr. 40. K tomuto znázornění byla použita funkce `seeMapF3bas` (nezobrazuje mapu s čísly taxelů). Pro znázornění včetně čísel přiřazených taxelů neuronům (které zde postrádá smysl, protože jsou zde uvedeny taxely bez očíslování) najde čtenář tuto mapu stejně jako všechny ostatní na příloženém CD.

Z pěti provedených simulací ve třech případech mapa splňovala požadovanou vlastnost zachování topologie.

6. Testování naučených neuronových sítí a interpretace výsledků



Obrázek 39. Pozice taxelů přední části trupu



SOM 11-May-2014

Obrázek 40. Simulace 3, neurony obarveny podle pozice jim přiřazených taxelů

Interpretace a zhodnocení této simulace

V této simulaci jsem chtěl ukázat, že se mapa na velké souvislé kůži dokázala hezky naučit (dobře odrážela topologii kůže). Tvarem není tato část kůže až tak vzdálená od obdélníkového tvaru. Simulace je proto podobná provedeným simulacím s umělými doteky (obdélníková kůže).

Receptivní pole je nastaveno jako jednotné, pokrývající celou kůži pro všechny neurony. Vzhledem k velikosti kůže (440 taxelů) už by stálo za to uvažovat o rozdělení na dvě menší receptivní pole. Tím by se měla zvýšit úspěšnost učení a měl by pravděpodobně stačit menší počet trénovacích epoch.

Tato simulace je také ukázkou toho, že na souvislé kůži s tvarem blízkým obdélníkovému, dokáže mapa velmi dobře odrážet topologii vstupní vrstvy ve výstupní vrstvě i po naučení na reálných dotecích. Tento výsledek posiluje hypotézu, že diskontinuity, které lze pozorovat v simulacích s více geometricky složitějšími částmi kůže (jako jsou ostatní části kůže robota iCub), jsou způsobeny geometrickými omezeními při reprezentaci těchto částí kůže (viz 6.3.4), nikoliv samotnou neuronovou sítí.

6.3.6. 4. simulace: homunculus

V této simulaci se mapa tvoří pro dlaň pravé ruky, pravé předloktí, nadloktí a celou přední část trupu. Tedy pro polovinu celé kůže robota iCub (až na kůži přední části trupu, která pokrývá obě poloviny těla). Na mapu lze pak nahlížet jako na senzoryckou mapu homuncula odpovídající jedné mozkové hemisféře.

Mapa má označení 0905p2.

Parametry simulace

Parametry simulace jsou:

```
o=10; p=25; %rozmery mapy
Mtren=[Bpalm(1:4000,:);BpalmForearmPrech(1500:2000,:);Bforearm;
BforearmUpperarmPrech([200:400,700:1000,2500:3000],:);BupperarmA;
BupperarmTorsoPrech([1:200,1000:1200,2000:2200,3000:3200,4000:4200,
5000:5200,6000:6200,7000:7200,8000:8200],:);
BtorsoA];
pocTrEp=5; %pocet trenovacich epoch-parametr trainlen
radiusIni=5; radiusFin=1; %radius init/final
```

Při vytvoření trénovací množiny bylo nutné zmenšit počet doteků emulujících souvislost kůže (matice s "Prech" na konci názvu). Jinak by příslušné krajní taxely byly moc často stimulované a zabíraly by na mapě moc neuronů. Pro vytvoření uvedené trénovací množiny je potřeba nejdříve načíst soubor `trenDataall.mat` (viz příložené CD). V tomto souboru jsou také pozice taxelů použitých v této simulaci.

Nastavení receptivních polí je takové, že neurony 1 až 30 mají ve svém receptivním poli dlaň a sousední okraj předloktí. Neurony 31 až 100 mají ve svém receptivním poli předloktí a sousední okraje dlaně a nadloktí. Neurony 101 až 170 mají ve svém receptivním poli nadloktí a sousední okraje předloktí a přední části trupu. Neurony 171 až 250 mají ve svém receptivním poli přední část trupu a sousední okraj nadloktí. Konvenci číslování neuronů si může čtenář připomenout v 5.1.1. Přesné nastavení receptivních polí může čtenář zjistit ze skriptu, který spouští simulaci s označením 0905p2 (viz příložené CD).

V kapitole C (příloha) jsou uvedeny všechny podstatné informace o tom, jak na sebe části kůže robota iCub navazují.

Znázornění

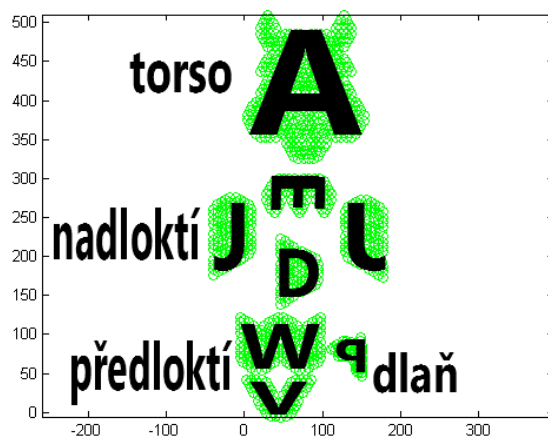
Při grafickém znázornění mapy budou jednotlivé pláty kůže označeny podle obr. 41.

Grafické znázornění jedné z takto naučených map (0905p2o3) je na obr. 42. V uvedeném grafickém znázornění jsem vyznačil hranice jednotlivých plátů kůže (značení viz obr. 41). Podle čar přechodů (viz popis u obrázku) lze určit, jak na sebe jednotlivé pláty v mapě navazují.

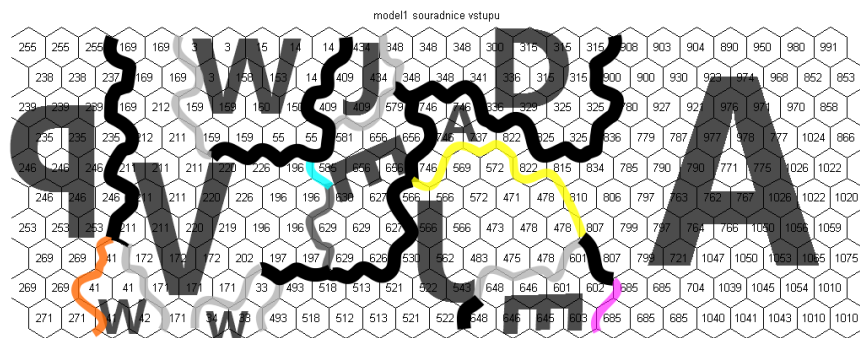
Všechny naučené mapy v této simulaci měly několik diskontinuit (tlusté černé čáry ve zmíněném obrázku). Domnívám se, že tvary jednotlivých částí kůže neumožňují jejich reprezentaci v obdélníkové mapě bez diskontinuit (viz 6.3.4). Proto nelze jednoznačně tvrdit, že by se mapy naučily špatně.

Interpretace a zhodnocení této simulace

Cílem této simulace bylo vytvořit senzoryckou mapu homuncula jedné mozkové hemisféry robota iCub. Z obr. 42 je vidět, že reprezentace částí kůže často na sebe navazují



Obrázek 41. Označení plátů kůže robota iCub



SOM 10-May-2014

Obrázek 42. Simulace 4, u neuronů jsou uvedena čísla jim přiřazených taxelů. Reprezentace plátů kůže jsou označeny symboly podle obr. 41. Barevnými čarami (žlutá, modrá, fialová, oranžová, tmavě šedá) jsou přibližně vyznačeny přechody mezi reprezentacemi částí kůže (předloktí, nadloktí...). Barvy přechodů mezi reprezentacemi částí kůže odpovídají barvám elips, které označují přechody částí z obr. 48 z přílohy C. Barevné čáry tedy označují přechod vytvořený koaktivacemi sousedních okrajů nenavazujících částí kůže. Pokud pláty kůže, které na sebe skutečně navazují (na iCubovi), na sebe navazují i v reprezentaci na mapě, jsou jejich reprezentace na obrázku odděleny světle šedou čarou. Tlusté černé čáry znázorňují skutečné diskontinuity.

tak, jak by na sebe měly navazovat podle jejich umístění na robotovi iCub. V mapě se také vyskytuje několik diskontinuit. Domnívám se, že z geometrických důvodů nelze všechny části kůže robota iCub reprezentovat v obdélníkové mapě bez vzniku diskontinuit (viz 6.3.4).

V této práci již nezbyl čas na hlubší zkoumání reprezentace částí kůže robota iCub v mapě. Proto tato simulace slouží spíše jako námět na další zkoumání. Do budoucna by bylo ještě vhodné vytvořit přehlednější způsob znázornění naučených map.

7. Závěr

Cílem této práce bylo vytvořit biologicky inspirovanou reprezentaci taktilních sensorů pomocí umělé neuronové sítě. Aplikační motivací vytvořeného modelu neuronové sítě je použití pro vnitřní reprezentaci taktilních sensorů humanoidního robota iCub.

Navržená neuronová síť musí ve výstupní vrstvě odrážet topologii taktilních sensorů. Tento požadavek vychází z topografického uspořádání reprezentace kůže celého těla v Brodmannově oblasti 3b primární somatosenzorické kůry v mozku. Další důležitou požadovanou vlastností je schopnost učení se z více současných od sebe vzdálených doteků. Tento požadavek vychází z pozorování, že je povrch lidského těla velmi často stimulován na více místech současně. Dále by měla neuronová síť umožňovat přednastavení oblastí kůže na které mají být jednotlivé neurony citlivé. Tímto způsobem je možné ovlivnit učení mapy. To může být výhodné v některých aplikacích experimentální robotiky nebo při zkoumání některých neurobiologických jevů. Další požadované vlastnosti lze najít v 2.3.

Nejdříve bylo vyzkoušeno použití samoorganizujících se map (SOM). Tento druh neuronové sítě, podle provedených experimentů, nesplňoval požadované vlastnosti. Proto byl modifikován. Zásadní modifikací byla změna ve způsobu určování vítěze. Při hledání vítěze byl místo normy rozdílu vstupního a váhového vektoru použit skalární součin zmíněných vektorů. Tím se podařilo výrazně změnit chování neuronové sítě.

S modifikovaným modelem SOM bylo provedeno přes 700 simulací. Nejdříve byly provedeny simulace s počítačově generovanými trénovacími množinami různých vlastností. Pro většinu vygenerovaných množin se podařilo najít parametry učení a map, při kterých mapy většinou splňovaly požadované vlastnosti. Překvapivým výsledkem bylo, že se mapa dokázala naučit a splnit požadované vlastnosti i při šesti současných simulacích kůže.

Omezený počet simulací byl proveden i na reálných datech z robota iCub. Při tvorbě těchto trénovacích dat byla kůže v každém okamžiku stimulována pouze na jednom místě. V případě, že se mapa vytvářela pro jednu souvislou část kůže (dlaň, přední část trupu), byly výsledky obdobné jako při simulacích s uměle vytvořenými doteky. To znamená, že se podařilo nalézt parametry učení, při kterých naučené mapy většinou splňovaly požadované vlastnosti.

Pokud se mapa vytvářela pro několik geometricky složitějších částí kůže, vznikaly v reprezentaci kůže na mapě diskontinuity. To se dalo očekávat, protože pravděpodobně není možné do obdélníku (tvar mapy) poskládat a zdeformovat objekty s tvarem jednotlivých částí kůže tak, aby na sebe navazovaly stejně, jako před deformací a nedotýkaly se na místech, kde na sebe před deformací nenavazovaly. To znamená, že se jedná spíše o problém geometrický než o problém neuronové sítě. V sensorické mapě reprezentace kůže v lidském mozku se vyskytuje několik diskontinuit, proto nelze tvrdit, že by diskontinuity v mapách ze simulací zabraňovaly použití je pro reprezentaci uvedených částí kůže. Z výsledků této práce nelze činit o tomto problému žádné seriózní závěry.

Při pokračování v práci na tomto tématu by bylo vhodné detailněji prozkoumat vliv nastavení jednotlivých parametrů na učení navržené neuronové sítě. Protože v této práci bylo nutné realizovat velké množství simulací a mnoho těchto simulací bylo časově náročných, nebylo možné jich provést dostatečný počet pro získání statistických údajů o

7. Závěr

učení. Proto by v budoucnu bylo vhodné provést jich větší počet a tyto statistické údaje získat. Dále by bylo zajímavé pokračovat v simulacích na reálných datech z robota iCub a prozkoumat, jak se vytváří mapa pro více geometricky složitějších částí kůže. Dalším předmětem zkoumání by mohlo být provedení učení při více současných dotecích na robotovi iCub.

Podle dosažených výsledků má uvedený model neuronové sítě dobré předpoklady pro budoucí použití k vnitřní reprezentaci taktilních senzorů humanoidních robotů i k použití v počítačových simulacích některých neurobiologických procesů.

Příloha A.

Algoritmus předzpracování taktilních vstupů

V této kapitole popisují algoritmus získání reprezentace dat s informacemi o stimulaci kůže, ve formátu vhodném pro zpracování v Matlabu, z dat přicházejících z kůže robota iCub [1]. Přicházející data s doteky jsou již předzpracována, nejedná se o nízkoúrovňová data. Celý algoritmus je realizován dvěma funkcemi v Matlabu. Obě uvedené funkce lze nalézt na přiloženém CD ve složce `predzpracovani`.

A.1. Formát vstupních dat z iCuba

Formát vstupních dat je detailněji popsán v [4]. Data, se kterými pracuji, jsou již ve vysokoúrovňové reprezentaci, kterou zajišťuje softwarový modul `skinManager` [13]. Data jsou ve formátu určeném `skinContactList` [14]. `skinContactList` je list `skinContact` [15].

Každý `skinContactList` má následující formát:

```
(SKIN_CONTACT_VECTOR_1) ... (SKIN_CONTACT_VECTOR_N)
```

Každý `SKIN_CONTACT_VECTOR` odpovídá jednomu shluku stimulovaných taxelů. Formát `SKIN_CONTACT_VECTOR` je následující:

```
((contactId bodyPartId linkNumber skinPart) (centerOfPressure_x cOP_y cOP_z)
(force_x f_y f_z) (moment_x m_y m_z) (geometricCenter_x gC_y gC_z)
(surfaceNormalDirection_x sND_y sND_z) (activatedTaxelId1 aTid2 .. aTidN)
averagePressure)
```

Jednotlivé proměnné mají tento význam:

- `bodyPartId`: označení části těla (TORSO=1, HEAD=2, LEFT_ARM=3, RIGHT_ARM=4, LEFT_LEG=5, RIGHT_LEG=6)
- `linkNumber`: číslo ramene vzhledem ke specifikované části těla
- `skinPart`: označení části kůže (SKIN_LEFT_HAND=1, SKIN_LEFT_FOREARM=2, SKIN_LEFT_UPPER_ARM=3, SKIN_RIGHT_HAND=4, SKIN_RIGHT_FOREARM=5, SKIN_RIGHT_UPPER_ARM=6, SKIN_FRONT_TORSO=7)
- `cOfPressure`: střed tlakového působení od doteku
- `geometricCenter`: geometrický střed dotýkané oblasti
- `surfaceNormalDirection`: směr normály k dotýkané oblasti
- `activatedTaxelId`: list aktivovaných taxelů
- `averagePressure`: průměrný výstup aktivovaných taxelů

Vstupní soubor vytváří `dataDumper` [16], který do vstupního souboru přidá do prvních dvou sloupců iterátor a časovou značku ([sekundy]). Řádek vstupního souboru pak má následující formát:

```
iterator timeStamp skinContactList
```

V případě, že v daný okamžik není aktivován žádný taxel, `skinContactList` se na řádku neobjeví.

A.2. Funkce předzpracování taktilních vstupů

Zpracování dat ze souboru se vstupními daty z `iCuba` až po maticovou reprezentaci jednotlivých doteků zajišťují dvě funkce. První z nich `data2actTax` zajišťuje vytěžení časové značky, `bodyPartId`, `linkNumber`, `skinPart` a `activatedTaxelId` ze vstupních dat z `iCuba`. Ostatní informace jsou zahozeny. Funkce následně uloží tyto informace do textového souboru. Druhá funkce `loadActTaxels` z výstupu funkce `data2actTax` zajistí vytvoření maticové reprezentace v `Matlabu`.

A.2.1. Funkce `data2actTax`

Funkci se zadá vstupní soubor s názvem `InputFile` ve formátu uvedeném v A.1. Funkce `data2actTax` uloží výstup do textového souboru s názvem `OutputFile`. Funkce se volá následovně:

```
data2actTax(InputFile,OutputFile)
```

Formát výstupního textového souboru `OutputFile` je následující:

- První řádek obsahuje časovou značku z prvního řádku souboru `InputFile`. Zbytek parametrů na řádku je formálně doplněn nulami. Konkrétně vypadá takto:
0 timeStamp 0 0 0 0
- Další řádky mají tento formát:
iterator timeStamp bodyPartId linkNumber skinPart activatedTaxelId1
activatedTaxelId2 .. activatedTaxelIdk

Příklad použití

Mám soubor `data.log` vstupních dat z `iCuba`. Obsah souboru je následující:

```
8511 1378906886.053001
8512 1378906886.073109
8513 1378906886.093132
8514 1378906886.113139
8515 1378906886.133298 ((136881 3 4 2) (-0.018 -0.041 -0.025)
(0.296042 -0.075339 0.369449)(0.0 0.0 0.0) (-0.018 -0.041 -0.025)
(-0.613 0.156 -0.765) (336) 9.6588)
8516 1378906886.153388 ((136882 3 4 2) (-0.019687 -0.042012 -0.024146)
(1.940303 -0.401482 1.830237) (0.0 0.0 0.0) (-0.019667 -0.042667 -0.024333)
(-0.708708 0.146644 -0.668506) (336 337 338) 18.252017)
((136907 1 2 7) (0.08 0.1 0.0) (-0.75191 -0.0 -0.0) (0.0 0.0 0.0)
(0.08 0.1 0.0) (1.0 0.0 0.0) (117 119) 15.038191)
8517 1378906886.173385 ((136883 3 4 2) (-0.019751 -0.042061 -0.024116)
(2.633877 -0.540827 2.458785) (0.0 0.0 0.0) (-0.019667 -0.042667 -0.024333)
(-0.712286 0.146257 -0.664935) (336 337 338) 24.651862)
```

Nyní v Matlabu zavolám:

```
data2actTax('data.log', 'dataSkin.txt')
```

Obsah souboru `dataSkin.txt` je pak následující:

```
0 1378906886.053001 0 0 0 0
1 1378906886.133298 3 4 2 336
2 1378906886.153388 3 4 2 336 337 338
2 1378906886.153388 1 2 7 117 119
3 1378906886.173385 3 4 2 336 337 338
```

A.2.2. Funkce `loadActTaxels`

Funkci se zadá vstupní soubor s názvem `InputFile`. Tento soubor je výstupní soubor funkce `data2actTax`. Funkce `loadActTaxels` ze vstupního souboru načte data a ve speciální maticové reprezentaci je uloží jako soubor formátu `mat-file`. Doteky jsou pak reprezentovány jako vektory. Každý vektor reprezentuje stav kůže (doteky) v určitý časový okamžik. Každé složce vektoru je přiřazen jeden taxel. Složky vektoru odpovídající aktivovaným taxelům, mají nastavenou hodnotu 1, ostatní složky mají nastavenou hodnotu 0. Vektory jsou seskupeny do řádků matice. V druhé matici, která má stejný počet řádků jako matice s vektory s doteky je uložen čas snímání vektorů s doteky od začátku snímání (první řádek ve vstupním souboru).

Některé složky vektorů s doteky mohou reprezentovat taxely, které nejsou využity (nejsou zapojené) nebo se používají pro jiný účel (např. měření teploty). Tyto složky vektorů mají vždy hodnotu 0 a je vhodné je před další prací s daty odstranit.

Offset částí kůže

Každý taxel kůže ze vstupního souboru je jednoznačně určen číslem části kůže (`skinPart`) a jeho identifikačním číslem `id` (v rámci dané `skinPart`). Identifikační číslo taxelu nabývá hodnot mezi 0 a $n - 1$, kde n je celkový počet taxelů dané části kůže. Protože potřebují reprezentovat taxel pouze jedním číslem v rámci celé kůže, zavedu offsety pro jednotlivé části kůže. Pro představu složky vektoru s číslem 1 až 192 (`skinPart` se 192 taxely) budou reprezentovat taxely na levé ruce (`skinPart` s číslem 1), složky s číslem 193 až 576 (`skinPart` s 384 taxely) budou reprezentovat taxely na levém předloktí (`skinPart` s číslem 2) atd. Každému taxelu na kůži přiřadím číslo

$$tn = \text{offset}(\text{skinPart}) + \text{taxelId}$$

Pro hodnotu offsetu platí

$$\text{offset}(i) = \sum_{j=1}^{i-1} \text{počet_taxelů_skinPart}(j) + 1$$

Hodnoty offsetů pro jednotlivé části kůže jsou shrnuty v Tab. 5.

Příklad použití

Pro usnadnění orientace jsou čísla o stejném významu obarveny stejnou barvou. Mám výstupní soubor `dataSkin.txt` z funkce `data2actTax`:

Tabulka 5. Tabulka s hodnotami offsetů pro jednotlivé části kůže

skin part	offset
1	1
2	193
3	577
4	1345
5	1537
6	1921
7	2689

```
0 1378906886.053001 0 0 0 0
1 1378906886.133298 3 4 2 336
2 1378906886.153388 3 4 2 336 337
2 1378906886.153388 1 2 1 117
```

V Matlabu zavolám:

```
loadActTaxels('dataSkin.txt', 'dataMatr.mat')
```

Po načtení souboru `dataMatr.mat` do Matlabu dostanu matici $B \in \{0, 1\}^{2 \times 3456}$. Hodnoty všech prvků matice B kromě dále uvedených jsou 0. Na těchto pozicích jsou nastaveny hodnoty 1.

$$B(1, offset(2) + 336) = B(1, 193 + 336) = 1$$

$$B(2, offset(2) + 336) = 1$$

$$B(2, offset(2) + 337) = 1$$

$$B(2, offset(1) + 117) = B(2, 1 + 117) = 1$$

Matice T obsahuje časy od začátku snímání doteků.

$$T = \begin{bmatrix} 1378906886.133298 - 1378906886.053001 \\ 1378906886.153388 - 1378906886.053001 \end{bmatrix} = \begin{bmatrix} 0.0803 \\ 0.1004 \end{bmatrix}$$

Příloha B.

mSOM v Matlabu

Funkce potřebné pro mSOM vznikly modifikací funkcí ze SOM toolboxu [17] určeného pro Matlab. Původní funkce SOM toolboxu zůstaly zachovány, pouze k nim byly přidány modifikované funkce. Modifikované funkce zde uvedu včetně stručného popisu. SOM toolbox s přidávanými funkcemi pro mSOM lze nalézt na přiloženém CD ve složce `somtoolboxDotProduct`. Protože modifikované funkce jsou velmi podobné funkcím SOM toolboxu budu se při jejich popisu odvolávat na jejich ekvivalenty v SOM toolboxu. Popis pak lze nalézt v online dokumentaci k SOM toolboxu [12].

B.1. Funkce pro učení mSOM

Funkce vznikla modifikací funkce `som_seqtrain` ze SOM toolboxu. Funkce pro učení mSOM se volá tímto způsobem:

```
sMap= som_seqtrainDPN(sMap, sD, RF, varargin)
```

Parametr `sMap` je struktura s informacemi o mapě, datová struktura `sD` obsahuje trénovací vektory a `varargin` představuje volitelné parametry. Význam uvedených parametrů funkce je totožný s jejich významem ve funkci `som_seqtrain`. Parametr `RF` představuje matici v jejíž řádcích jsou vektory receptivních polí. Rozměr matice `RF` je stejný jako rozměr matice s váhovými vektory. Vektor receptivního pole v *i*-tém řádku náleží stejnému neuronu jako váhový vektor v *i*-tém řádku.

Část skriptu realizující učení pak může vypadat třeba takto:

```
sD=som_data_struct(M); %vytvořeni dat. struktury, M-matice s trénovacími vektory
sM=som_randinit(sD,'msize',[10 10]); %náhodná inicializace, velikost mapy 10*10
sM=som_seqtrainDPN(sM,sD,RF,'trainlen',6); %trainlen-počet trénovacích epoch
```

Jádro modifikace spočívá v následujících částech kódu:

```
for(...){ %trenovaci cyklus

%find BMU
x = D(samples(ind),:); % pick one sample vector
known = ~isnan(x); %known components (solve problems with nans)
Dx = M(:,known) - x(mu_x_1,known); % each map unit minus the sample vector
Mpom=M(:,known).*RF(:,known); %RF..receptive fields
pom=Mpom*x(1,known)'; %dot product with each map unit
[qerr bmu] = max(pom); %find maximum
..
%update M
M(:,known) = M(:,known) - h(:,ones(sum(known),1)).*Dx;
```

```

%normalize
for i=1:1:size(M,1)
M(i,:)=(1/sqrt(M(i,:)*M(i,:)')).*M(i,:);
end

} %konec trenovaciho cyklu
M=M.*RF; %set zeros in the M where in the RF are zeros

```

B.2. Funkce pro určení nejsilněji pálicího neuronu

Mějme naučenou mSOM ve struktuře sM. Pak mohu potřebovat zjistit, který neuron v mapě bude nejsilněji pálit na určitý dotek ve vstupním vektoru. K tomuto účelu slouží funkce `som_bmusDP`. Funkce vznikla modifikací funkce `som_bmus` ze SOM toolboxu.

Mějme matici T v jejíchž řádcích jsou vstupní vektory, pro které nás zajímají nejsilněji pálicí neurony. Funkce se pak volá takto:

```
[Bmus,DP] = som_bmusDP(sM,T)
```

V matici Bmus jsou v řádcích vítězné neurony příslušející vstupním vektorům v řádcích matice T. V řádcích matice DP jsou hodnoty skalárních součinů vstupních vektorů s vítěznými neurony.

V případě, že je potřeba zjistit i další pálicí neurony v pořadí, lze funkci volat tímto způsobem:

```
[Bmus,DP] = som_bmusDP(sM,T,'all')
```

Místo matice T lze použít z ní vytvořenou datovou strukturu (`som_data_struct`).

B.3. Funkce pro určení nejsilněji pálicích neuronů s laterální inhibicí

Mějme naučenou mSOM ve struktuře sM. V případě více doteků ve vstupním vektoru budu chtít zjistit více neuronů, které silně pálí. Dále mohu chtít, aby ideálně každý z doteků ve vstupním vektoru byl reprezentován pouze jedním pálicím neuronem. Proto využiji laterální inhibice, která je popsána včetně algoritmu v 4.6. K určování pálicích neuronů s využitím principu laterální inhibice slouží funkce `som_bmusDPLatInh`.

Mějme matici T v jejíchž řádcích jsou vstupní vektory, pro které nás zajímají nejsilněji pálicí neurony. Funkce se pak volá takto:

```
[Bmus,d] = som_bmusDPLatInh(sM,T)
```

V matici Bmus jsou v řádcích vítězné neurony příslušející vstupním vektorům v řádcích matice T. V řádcích matice d jsou hodnoty intenzity pálení jednotlivých neuronů (hodnoty skalárních součinů po laterální inhibici). Funkce také automaticky vykreslí graf s intenzitou pálení jednotlivých neuronů. Místo matice T lze použít z ní vytvořenou datovou strukturu (`som_data_struct`).

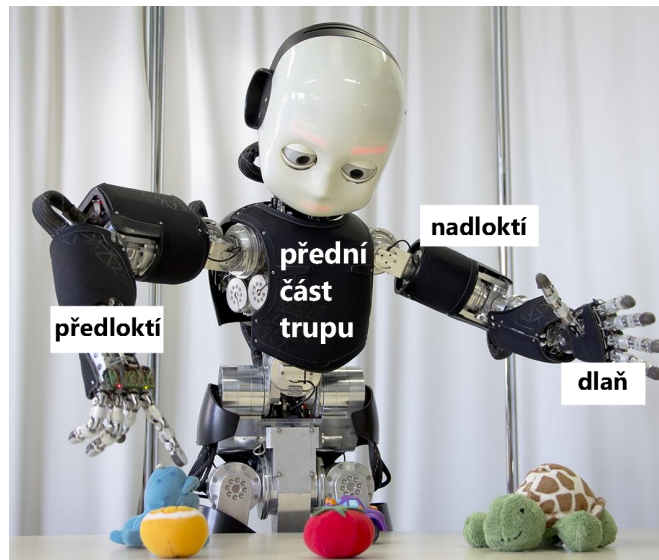
Příloha C.

Kůže robota iCub

V této kapitole popíšu kůži robota iCub z pohledu geometrie (nikoliv vnitřní stavby).

C.1. Části kůže robota iCub

Kůže robota iCub se skládá z více částí. Konkrétně z částí pro dlaně, předloktí, nadloktí a přední část trupu (torso). Každá z těchto částí se skládá z tzv. plátů kůže (patches). Na obr. 43 je robot iCub s popisky částí kůže.



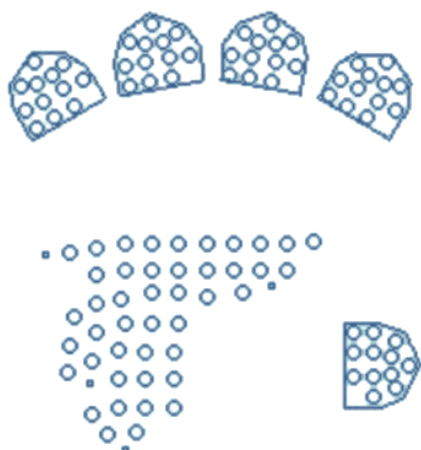
Obrázek 43. Robot iCub s popisky částí kůže (foto Laura Taverna)

C.1.1. Dlaň

Kůže dlaně se skládá z plátu pro samotnou dlaň. Dále na každém konečku prstu je několik taxelů. V této práci se budu zabývat pouze samotnou dlaní. Na obr. 44 je znázorněna kůže dlaně a konečků prstů.

C.1.2. Předloktí

Kůže předloktí se skládá ze dvou plátů kůže. Struktura kůže je znázorněna na obr. 45. Podrobnější popis lze nalézt na uvedeném obrázku. Na obrázku je také znázorněno jak na sebe pláty navazují.



Obrázek 44. Kůže dlaně. Na obrázku lze vidět dlaň a konečky prstů

C.1.3. Nadloktí

Kůže nadloktí se skládá ze čtyř plátů kůže. Struktura kůže je znázorněna na obr. 45. Podrobnější popis lze nalézt na uvedeném obrázku. Na obrázku je také znázorněno jak na sebe pláty navazují.

C.1.4. Přední část trupu

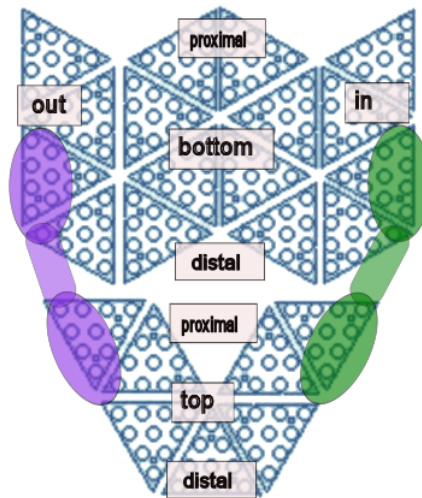
Kůže přední části trupu se skládá z jednoho plátu kůže. Struktura kůže je znázorněna na obr. 45. Podrobnější popis lze nalézt na uvedeném obrázku.

C.2. Návaznost částí kůže

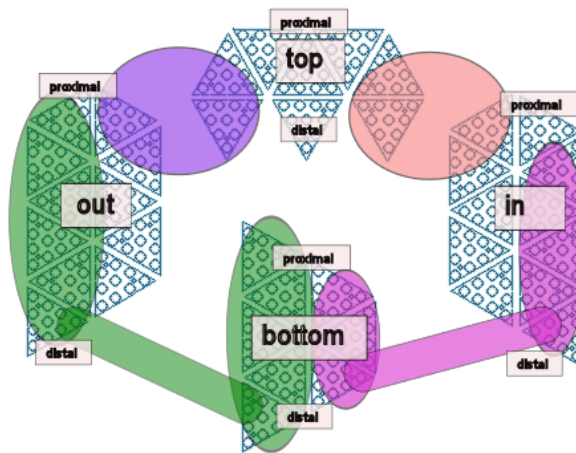
Jak na sebe navazují pláty částí kůže je popsáno v C.1.

Některé blízké části kůže robota iCub na sebe ve skutečnosti nenavazují (např. předloktí a nadloktí je odděleno kloubem, který není pokryt kůží). To lze řešit současnou stimulací sousedních okrajů nenavazujících okrajů kůže. Mapa si koaktivace okrajů interpretuje, jako kdyby na sebe obě části kůže navazovaly.

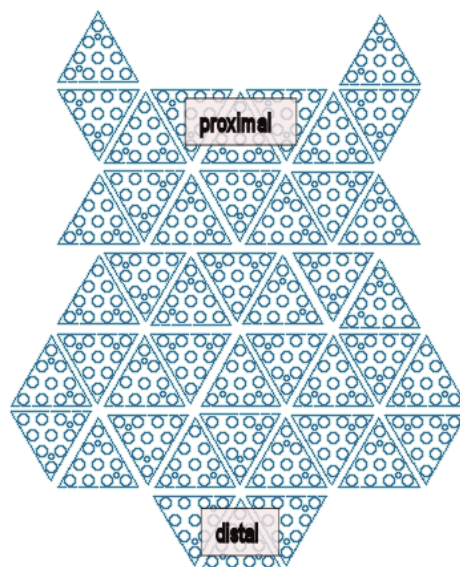
Z obr. 48 si může čtenář udělat přibližnou představu o tom, které nenavazující části kůže jsou koaktivovány, aby mapa "získala dojem", že na sebe navazují. Obrázek s očíslovanými pozicemi taxelů nelze kvůli jeho velikosti zde uvést. Čtenář může nalézt tento obrázek (s možností zoomu) v souboru `homunkulusTaxPos.fig` (viz příložené CD). Pokud by se čtenář rozhodl sám detailně zkoumat naučenou mapu, má již všechny důležité informace o stimulovaných částech kůže robota iCub.



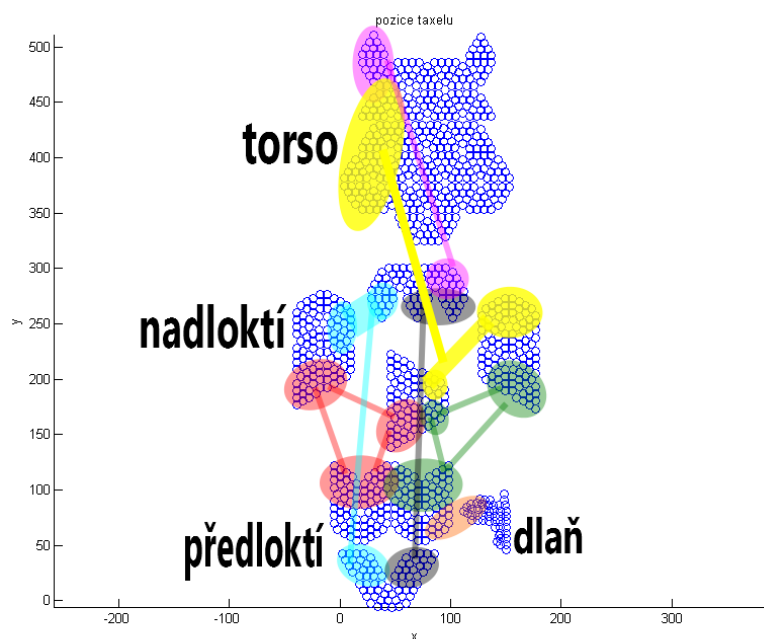
Obrázek 45. Kůže předloktí. Plát kůže ve tvaru V je na horní části předloktí. Prostřední část plátu ve tvaru W je na spodní části předloktí. Krajní části plátu kůže ve tvaru W jsou po stranách předloktí. Části plátů s popiskem proximal jsou blíže rameni (více nahoře), části s popiskem distal jsou dále od ramene (více dole). Části plátů uvnitř elips stejné barvy na sebe na iCubovi navazují.



Obrázek 46. Kůže nadloktí. Plát s popiskem top je na horní části paže, plát s popiskem bottom je na spodní části paže. Plát s popiskem out je na vnější straně paže, plát s popiskem in je na vnitřní straně paže. Části plátů s popiskem proximal jsou blíže rameni (více nahoře), části s popiskem distal jsou dále od ramene (více dole). Části plátů uvnitř elips stejné barvy na sebe na iCubovi navazují.



Obrázek 47. Kůže přední části trupu. Část kůže s popiskem proximal je u ramene.



Obrázek 48. V obrázku jsou vyznačeny oblasti kůže, které jsou koaktivovány pro emulaci jejich vzájemné návaznosti. Pokud jsou dvě oblasti kůže uvnitř propojených elips stejné barvy, jsou tyto oblasti také současně stimulovány za účelem zajištění jejich plynulého přechodu v reprezentaci na mapě

Příloha D.

Obsah příloženého CD

Struktura obsahu je následující (složky jsou tučně):

funkce složka obsahuje funkce uvedené v práci

- **ostatni**
- **predzpracovani**
- **somtoolbox05DotProduct**
- **vizualizace**

simulace složka obsahuje soubory spojené se simulacemi z práce

- **iCub**
 - * skriptySimulaciCub.txt
 - * **simulace1**
 - * **simulace2**
 - * **simulace3**
 - * **simulace4**
- **umele**
 - * skriptySimulaciUmele.txt
 - * **serie1**
 - * **serie2**
 - * **serie3**
 - * **serie4**
 - * **serie5**
 - * **serie6**

BCprace složka obsahuje elektronickou verzi bakalářské práce

Literatura

- [1] Giorgio Metta et al. “The iCub humanoid robot: An open-systems platform for research in cognitive development”. In: *Neural Networks* 23 (8 2010), s. 1125–1134.
- [2] Rolf Pfeifer a Scheier Christian. *Understanding intelligence*. MIT press, 2001.
- [3] Eric R. Kandel, James H. Schwartz a Thomas M. Jessell. *Principles of neural science*. fourth edition. 2000.
- [4] A. Schmitz et al. “Methods and Technologies for the Implementation of Large-Scale Robot Tactile Sensors”. In: *Robotics, IEEE Transactions on* 27 (3 2011), s. 389–400.
- [5] Martha J. Farah. “Why Does the Somatosensory Homunculus Have Hands Next to Face and Feet Next to Genitals? A Hypothesis”. In: *Neural computation* 10 (8 1998), s. 1983–1985.
- [6] Tom Stafford a Stuart Wilson. “Self-organisation can generate the discontinuities in the somatosensory map”. In: *Neurocomputing* 70 (10-12 2007), s. 1932–1937.
- [7] Teuvo Kohonen. “The Self-Organizing Map”. In: *Proceedings of the IEEE* 78 (9 1990), s. 1464–1480.
- [8] Teuvo Kohonen. “Physiological interpretation of the Self-Organizing Map algorithm”. In: *Neural Networks* 6 (7 1993), s. 895–905.
- [9] Juha Vesanto. *SOM algorithm implementation in SOM Toolbox*. URL: <http://www.cis.hut.fi/somtoolbox/documentation/somalg.shtml> (cit. 2014).
- [10] Simon Haykin. *Neural Networks: A Comprehensive Foundation*. second edition. 1999.
- [11] Igor Farkaš. “Samoorganizujúce se mapy”. In: Vladimír Kvasnička a kolektiv. *Úvod do teórie neurónových sietí*. 1997.
- [12] *online dokumentace SOM Toolbox*. URL: <http://www.cis.hut.fi/projects/somtoolbox/package/docs2/somtoolbox.html> (cit. 2014).
- [13] *iCub-software module skinManager*. URL: http://wiki.icub.org/iCub/main/dox/html/group__icub__skinManager.html (cit. 2014).
- [14] *iCub-skinContactList Class Reference*. URL: http://wiki.icub.org/iCub/main/dox/html/classiCub_1_1skinDynLib_1_1skinContactList.html (cit. 2014).
- [15] *iCub-skinContact Class Reference*. URL: http://wiki.icub.org/iCub/main/dox/html/classiCub_1_1skinDynLib_1_1skinContact.html (cit. 2014).
- [16] *iCub-dataDumper*. URL: http://wiki.icub.org/iCub/main/dox/html/group__dataDumper.html (cit. 2014).
- [17] Juha Vesanto et al. *SOM toolbox for Matlab 5*. Citeseer, 2000.