



**KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS**

Tomas Šumskas

**KONCEPTAIS GRĮSTOS IŠMANIOSIOS TEKSTO ĮVEDIMO
SISTEMOS TYRIMAS**

Baigiamasis magistro projektas

Vadovas

Prof. dr. Robertas Damaševičius

KAUNAS, 2015

KAUNO TECHNOLOGIJOS UNIVERSITETAS

INFORMATIKOS FAKULTETAS

**KONCEPTAIS GRĮSTOS IŠMANIOSIOS TEKSTO ĮVEDIMO
SISTEMOS TYRIMAS**

Baigiamasis magistro projektas
Programų sistemų inžinerija (kodas 621E16001)

Vadovas

Prof. dr. Robertas Damaševičius
2015-05-25

Recenzentas

Doc. dr. Sigitas Drąsutis
2015-05-25

Projektą atliko

Tomas Šumskas
2015-05-25



KAUNO TECHNOLOGIJOS UNIVERSITETAS

Informatikos

(Fakultetas)

Tomas Šumskas

(Studento vardas, pavardė)

Programų sistemų inžinerija (kodas 621E16001)

(Studijų programos pavadinimas, kodas)

Baigiamojo projekto „Konceptais grįstos išmaniosios teksto įvedimo sistemos tyrimas“

AKADEMINIO SAŽININGUMO DEKLARACIJA

20 15 m. Gegužės 25 d.
Kaunas

Patvirtinu, kad mano, Tomo Šumsko, baigiamasis projektas tema „Konceptais grįstos išmaniosios teksto įvedimo sistemos tyrimas“ yra parašytas visiškai savarankiškai ir visi pateikti duomenys ar tyrimų rezultatai yra teisingi ir gauti sąžiningai. Šiame darbe nei viena dalis nėra plagijuota nuo jokių spausdintinių ar internetinių šaltinių, visos kitų šaltinių tiesioginės ir netiesioginės citatos nurodytos literatūros nuorodose. Įstatymų nenumatytų piniginių sumų už šį darbą niekam nesu mokėjęs.

Aš suprantu, kad išaiškėjus nesąžiningumo faktui, man bus taikomos nuobaudos, remiantis Kauno technologijos universitete galiojančia tvarka.

(vardą ir pavardę įrašyti ranka)

(parašas)

TURINYS

Paveikslėlių sąrašas:	6
Lentelių sąrašas:	6
1. Įžanga	8
1.1. Dokumento paskirtis	8
1.2. Santrauka.....	8
2. Analitinė dalis	8
2.1. Tikslas	8
2.2. NCI ir BCI palyginimas	9
2.2.1. NCI.....	9
2.2.2. BCI.....	10
2.3. HBCI.....	11
2.4. Problemos sprendimas pasaulyje	11
2.4.1. DTU BCI Speller programa	12
2.4.2. Singapūro IR instituto P300 BCI	13
2.4.3. Dviejų dimensijų garsinė P300 teksto įvedimo sistema su teksto nuspėjimu	13
2.4.4. ERP grįsta lytėjimo teksto įvedimo sistema	13
2.4.5. Rezultatų palyginimas	14
2.4.6. Programų palyginimas	15
3. Siūlomas metodas	16
3.1. Įvadas	16
3.2. Konceptų panaudojimas	16
3.3. Vizualinės kalbos modeliavimas.....	17
4. Projektinė dalis.....	19
4.1. Projekto reikalavimai	19
4.1.1. Funkciniai reikalavimai.....	19
4.1.2. Nefunkciniai reikalavimai.....	19
4.2. Programų sistemos funkcijos	20
4.3. Architektūros tikslai	21
4.4. Sistemos statinis vaizdas	21
4.5. Sistemos dinaminis vaizdas.....	22
4.5.1. Koncepto išsirinkimas	22
4.5.2. Pasiūlymo generavimas ir atvaizdavimas	23
4.5.3. Naujo koncepto saugojimas	24
4.6. Duomenų vaizdas	25
4.7. Testavimas.....	25

5.	Eksperimentinė dalis	26
5.1.	Tyrime naudotų teksto įvedimo sistemų aprašymas	26
5.1.1.	Tradicinė teksto įvedimo sistema.....	26
5.1.2.	Konceptais grįsta teksto įvedimo sistema	27
5.2.	Eksperimento rezultatai.....	28
5.2.1.	Eksperimentas su tradicine teksto įvedimo sąsaja	28
5.2.2.	Eksperimentas su konceptais grįsta teksto įvedimo sąsaja	31
6.	Išvados	33
7.	Literatūra.....	33
8.	Terminų ir santrumpų žodynas.....	35
9.	Priedai	36
9.1.	Straipsniai.....	40
9.1.1.	Evaluation of user fatigue in neural computer interface system	40
9.1.2.	Development of Concept-Based EMG-Based Speller	44

Paveikslėlių sąrašas:

Pav. 1 Klaviatūros pavyzdys	9
Pav. 2 Elektrodo išdėstymas [1]	10
Pav. 3 Simbolių matrica	10
Pav. 4 Elektrodo vardai ir išdėstymas	11
Pav. 5 Grafinės sąsajos vaizdas su aktyvia simbolių įvedimo vieta (kairėje) ir aktyvia žodyno vieta (dešinėje)	12
Pav. 6 Teksto įvedimo programos sąsajos vaizdas	13
Pav. 7 Brailio stimulatorius naudotas pirštų galų stimuliuoti	14
Pav. 8 Raidžių ir pirštų sąryšis	14
Pav. 9 Teigiamą emociją išreiškiantis paveikslukas	17
Pav. 10 Pagrindinės vizualinės – konceptų – kalbos dalys	17
Pav. 11 Panaudos atvejų diagrama	20
Pav. 12 Sistemos išdėstymo diagrama	21
Pav. 13 Koncepto išsirinkimo veiklos diagrama	22
Pav. 14 Pasiūlymo generavimo ir atvaizdavimo diagrama	23
Pav. 15 Naujo koncepto saugojimo veiklos diagrama	24
Pav. 16 Duomenų bazės esybių diagrama	25
Pav. 17 Tradicinės teksto įvedimo sąsajos vaizdas	26
Pav. 18 Konceptai grįstos teksto įvedimo sąsajos vaizdas	27
Pav. 19 Grįžimo į šakninį lygį konceptas	27
Pav. 20 Konceptų medžio struktūra	28
Pav. 21 Tradicinės sąsajos tikslumas	30
Pav. 22 Tradicinės sąsajos bitų perdavimo greitis	30
Pav. 23 Tradicinės sąsajos įvedimo greitis	31
Pav. 24 Konceptų formuluotės pavyzdys	31
Pav. 25 Konceptais grįstos sąsajos įvedimo greitis	32

Lentelių sąrašas:

Lentelė 1 Rezultatų palyginimas	14
Lentelė 2 Programų palyginimas	15
Lentelė 4 Pavyzdinės vizualinės kalbos frazės	18
Lentelė 5 Tradicinės teksto įvedimo sąsajos eksperimento rezultatai	29
Lentelė 6 Konceptai ir jų reikšmės	36

Šumsko Tomo „Konceptais grįstos išmaniosios teksto įvedimo sistemos tyrimas“. Programų sistemų inžinerijos magistro baigiamasis projektas / vadovas prof. dr. Robertas Damaševičius; Kauno technologijos universitetas, informatikos fakultetas.

Kaunas, 2015. 54 p.

SUMMARY

This work is about the evaluation of concept based smart speller. Analysis of existing spellers is in the beginning. It also contains evaluation of these text input software. Next chapter describes the proposed method of concept use. It defines the motivation behind the creation of this kind of speller and characterizes modeling of visual language. Further main aspects of the system, like requirements, static and dynamic views of the systems and data view, are described. In the experiment chapter performed evaluation of concept based speller and its results are described in detail. Finally conclusions of this work are presented with the list of used literature sources. At the end of this document papers written during master studies can be found.

1. Įžanga

1.1. *Dokumento paskirtis*

Šis dokumentas yra programų sistemų inžinerijos studijų baigiamasis magistro darbas. Jis skirtas aprašyti studijų metu sukurtą sistemą, nusakant jos panaudojimo sritis, aprašant architektūrinius bei funkcijas. Taip pat šiame dokumente pateikiamas su sistema atliktas tyrimas ir jo išvados.

1.2. *Santrauka*

Šiame darbe atliekamas konceptais grįstos išmaniosios teksto įvedimo sąsajos tyrimas. Pradžioje išanalizuojami pasaulyje egzistuojantys sprendimai. Taip pat atliekamas šių sprendimų palyginimas. Tolesniame skyriuje aprašomas siūlomas konceptų panaudojimo metodas. Pateikiama motyvacija tokios įvedimo sąsajos kūrimui bei apibūdinama sumodeliuota vizualinė kalba. Toliau apžvelgiami esminiai sistemos įgyvendinimo aspektai: reikalavimai, statinis, dinaminis sistemos vaizdai bei duomenų vaizdas. Eksperimentinėje dalyje detaliai aprašomas atliktas eksperimentas bei gauti rezultatai. Galiausiai pateikiamos magistrinio darbo išvados ir literatūros sąrašas. Prieduose galima rasti darbo metu parašytus straipsnius.

2. Analitinė dalis

Nervų-kompiuterio sąsaja (NKS) gali padėti žmonėms turintiems stiprias negalias, tokias kaip paralyžius, šoninė amiotrofinė sklerozė ir panašios. Žmonės, prikaustyti prie vežimėlio ar lovos ir negalintys šnekėti, tačiau galintys bent šiek tiek pajudinti kokius nors raumenis NKS pagalbą gali perduoti signalus kompiuteriui, kur šie signalai pasitelkus programinę įrangą atliktų tam tikrą veiksmą.

Tačiau yra ir kitų nervų-kompiuterio sąsajos panaudojimo sričių: ją galima panaudoti mokymuisi. Vaikai galėtų išmokti naujų sudėtingų žodžių rašydami juos su NKS. Taip pat sąsaja galima panaudoti pramogai, pavyzdžiui valdyti televizorių ar kompiuterį, bei žaidimams, pavyzdžiui valdote paukštį, kuris skrenda tiesiai, tačiau po truputį leidžiasi, sujudinate raumenį ir paukštis suplasnoja – pakildamas aukščiau, taip jį valdydamas turite nusukti kuo toliau vengdami įvairių kliūčių.

2.1. *Tikslas*

Šio darbo tikslas yra sukurti išmaniąją teksto įvedimo sistemą, kurioje teksto įvedimo greitis būtų didesnis, o klaidų skaičius būtų mažesnis lyginant su analogiškoms sistemoms dėka naudojamų dirbtinio intelekto metodų (automatinis klaidų ištaisymas, įvedamos raidės/žodžio spėjimas, automatinis žodžio užbaigimas ir konceptų pasiūlymas). Taip programos vartotojas galės pranešti aplinkiniams apie savijautą, išreikšti pageidavimus ar tiesiog pasišnekėti.

Kadangi sunkių ligonių, kurie galėtų pasinaudoti šia sąsaja artimoje aplinkoje nėra, todėl darbas yra prototipinis ir labiau orientuotas sveikų žmonių pramogai.

2.2. NCI ir BCI palyginimas

2.2.1. NCI

Šios sąsajos efektyvumas priklauso nuo vartotojo sugebėjimo valdyti raumenis, tačiau nereikalauja didelio susikaupimo ir norint išmokti naudotis šia sąsaja efektyviai reikia mažiau laiko.

2.2.1.1. Teksto įvedimas su NCI

Dažniausiai teksto įvedimui su nervų-kompiuterio sąsaja pasirenkama judesiais valdyti pelės žymeklį, kuriuo iš klaviatūros, panašios į mobiliųjų telefonų, renkamas tekstas [1]. Sujungti atskirą judesį su individualia raide būtų per daug sudėtingas uždavinys.

Žemiau parodytas klaviatūros pavyzdys (žr. Pav. 1), ją sudaro 3 x 5 matrica, kurioje pirmose trejose eilutėse yra Romėniškos ir Korėjietiškos raidės, toliau eina ištrynimo (←) ir tarpo (→) simboliai, kalbos perjungimo mygtukas, įvedimo bei atšaukimo mygtukai.



Pav. 1 Klaviatūros pavyzdys

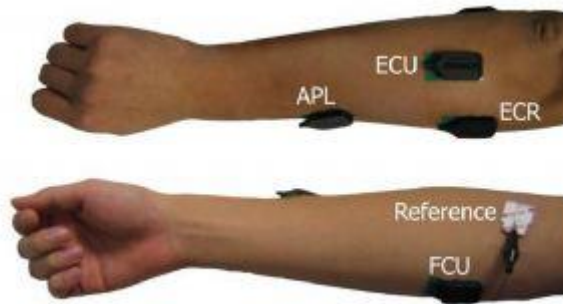
2.2.1.2. NCI signalai

Elektromiogramos metodu fiksuojamų raumenų sujudėjimo dažnis yra nuo 7 iki 20 Hz. Taip pat priklausomai nuo raumens iš kurio fiksuojamas signalas įtampos pokytis gali kisti nuo 50 μ V iki 20 mV.

2.2.1.3. NCI signalų šaltiniai

Elektrodai tvirtinami ant vartotojo raumenų. Tada elektromiografijos arba elektroneurografijos metodu fiksuojami raumenų ar jų skaidulų sukuriama elektriniai signalai.

Elektrodai gali būti tvirtinami prie bet kokių raumenų, tačiau kai kurie raumenys gali būti per maži ir jų generuojamų signalų nepakaks išskirti judesį iš triukšmo. Changmok Choi su kolegomis savo darbe pasirinko registruoti vartotojo riešo judesius [1], kuriais būtų valdomas pelės žymeklis. Jų pasirinktas elektrodų išdėstymas matomas Pav. 2.



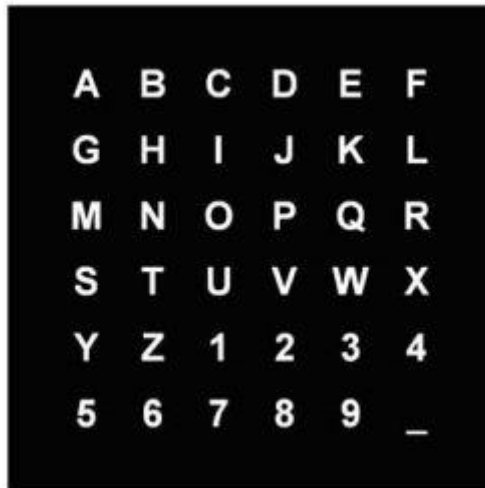
Pav. 2 Elektrođų išdėstymas [1]

2.2.2. BCI

Nepriklauso nuo nervų ar raumenų veiklos [2], būklės ar sugebėjimo jais naudotis, tačiau reikalauja nemažo susitelkimo ties sąsajos valdymu. Taip pat laikas – reikalingas apmokymui yra didesnis palyginus su NCI.

2.2.2.1. Teksto vedimas su BCI

Teksto įvedimo užduočiai su smegenų-kompiuterio sąsaja pasirenkama standartinė 6 x 6 simbolių matrica Pav. 3, su Anglų kalbos abėcėlės raidėmis, skaičiais nuo 1 iki 9 bei tarpo simboliu. Vartotojas sukonzentruoja dėmesį į norimą simbolį, tada matricos eilutės bei stulpeliai yra atsitiktine tvarka paryškunami. Iš vartotojo gautos reakcijos yra suklasifikuojamos ir identifikuotas simbolis atvaizduojamas ekrane [3].



Pav. 3 Simbolių matrica

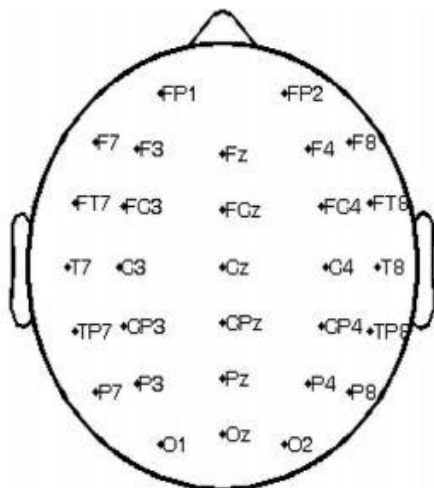
Taip pat gali būti naudojama ir 3 x 3 matrica tik su skaičiais, tačiau šiuo atveju naudojamas garsinis, o ne vaizdinis stimulus vartotojo reakcijai išgauti [4].

2.2.2.2. BCI signalai

Daugumos suaugusių žmonių elektroencefalogramoje fiksuojamos μ ir β smegenų bangos kinta atitinkamai nuo 8 iki 12 Hz (μ) ir nuo 18 iki 26 Hz (β) dažniais [5] bei α bangos kintančios nuo 7 iki 14 Hz.

2.2.2.3. BCI signalų šaltiniai

Šios sąsajos elektrodai tvirtinami ant vartotojo galvos specifinėse vietose, pavyzdžiui Tianyou Yu su kolegomis iš Pietų Kinijos technologijų universiteto tvirtino 30 elektrodų žemiau esančiame paveiksliuke pavaizduotose vietose [6]. Dažniausiai pasirenkama tarptautinė 10-20 elektrodų tvirtinimo ant galvos sistema, matoma Pav. 4. Elektrodų išdėstymas priklauso nuo to kokios smegenų bangos registruojamos.



Pav. 4 Elektrodų vardai ir išdėstymas

2.3. HBCI

Pastaruoju metu padaugėjo tyrimų susijusių su hibridinėmis smegenų-kompiuterio sąsajomis. Jų tikslas yra apjungiant keletą sprendimų (P300, SSVEP, N2, atskirų smegenų bangų aptikimas ir klasifikavimas ir raumenų signalų aptikimas ir klasifikavimas) išgauti geresnius rezultatus bei sumažinti laiką, reikalingą apmokyti vartotojui.

Pavyzdžiui Yuanqing Li ir kolegės panaudojo P300 potencialą ir SSVE potencialus vežimėlio valdymui [7]. Eksperimento rezultatų analizė parodė, jog apjungiant šiuos du sprendimus stipriai pagerino smegenų-kompiuterio sąsajos veikimo greitį ir tikslumą lyginant su įprastais BCI.

Robert Leeb pasiūlytas variantas [8] apjungiant raumenų ir smegenų signalus leistų vartotojams persijunginėti tarp vieno ar kito, pavyzdžiui pavargus raumenims ar sumažėjus koncentracijai būtų galima leisti pailsėti vienai daliai kol kita dirba. Tai kur kas labiau išplečia praktinį BCI sistemų efektyvumą ir panaudojamumą.

2.4. Problemos sprendimas pasaulyje

Naudojamų prietaisų ir metodų signalų apdorojimui efektyvumui dažnai pasitelkiama šabloninė teksto įvesties programa „Speller“. Jos pagalba nustatomos tokios sistemos charakteristikos kaip duomenų perdavimo greitis, vidutinis simbolių kiekis per minutę bei laikas reikalingas apmokyti vartotoją.

Programos grafinėje gali būti pavaizduotos įvairios simbolių matricos, 4 x 10 [9], 3 x 3 [10], 6 šešiakampiai ar net pilna QWERTY klaviatūra [11], tačiau dažniausiai pasirenkama 6 x 6 simbolių matrica

dėl savo universalumo. Joje galima sutalpinti anglų abėcėlės raidės, skaičius nuo 1 iki 9 bei tarpo simbolį, tačiau galimi nedideli nukrypimai, pavyzdžiui abėcėlė parinkta pagal vartotojų gimtąją kalbą. Tipinis grafinės sąsajos vaizdas matomas Pav. 3.

Šiai analizei pasirinktos keturios šabloninės teksto įvedimo programos.

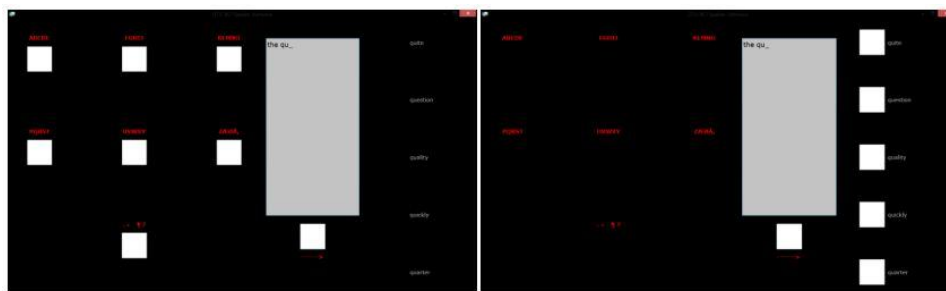
2.4.1. DTU BCI Speller programa

Danijos technikos universiteto darbuotojai sukūrė teksto įvedimo programą pagrįstą SSVE potencialais ir turinčią žodyną.

Devyni sveiki vartotojai buvo po vieną pasodinti tamsiame kambaryje 60 centimetrų atstumu nuo kompiuterio ekrano, kuriame rodomi vaizdiniai stimulai.

Signalų gavimui naudoti 3 paausuoti elektrodai, kurie buvo pritvirtinti prie vartotojo galvos.

Grafinė sąsaja (Pav. 5) susideda iš dviejų vietų su mirksinčiais taikiniais. Tarp tų vietų yra laukas kuriame matomas įvedinėjamas tekstas. Po šiuo lauku yra dar vienas visuomet aktyvus taikiny, kurį pasirinkus perjungiama aktyvi pusė. Kairėje pusėje yra septyni taikiniai su individualiais simboliais. Pirmuoju bandymu vartotojas išsirenka simbolių grupę, o antrajame išsirenkamas norimas simbolis. Dešinėje pusėje yra žodynas, iš kurio gali būti pasirinktas norimas žodis.



Pav. 5 Grafinės sąsajos vaizdas su aktyvia simbolių įvedimo vieta (kairėje) ir aktyvia žodyno vieta (dešinėje)

2.4.2. Singapūro IR instituto P300 BCI

Singapūro IR institute Chaunchu Wang, Cuntai Wang ir Haihong Zhang [9] pasitelkdami P300 potencialus sukūrė parodomąją teksto įvedimo programą.

Grafinę sąsają (Pav. 6) sudaro viršuje esantis teksto laukas ir po esanti 4 x 10 simbolių matrica.



Pav. 6 Teksto įvedimo programos sąsajos vaizdas

Vartotojų stimuliavimui atsitiktine tvarka vienas po kito mirksi mygtukai. Intervalas tarp mirkstelėjimų gali būti labai trumpas – apie 30 milisekundžių. Tačiau epochos duomenis reikalingi pamatyti P300 šabloną trunka nuo 150 iki 500 milisekundžių.

2.4.3. Dviejų dimensijų garsinė P300 teksto įvedimo sistema su teksto nuspėjimu

Martijn Schreuder su kolegomis [4] sukūrė dviejų dimensijų garsinę teksto įvedimo sistemą. Stimulai jų programoje kinta aukščio dimensijoje: aukštas, vidutinis, žemas garsas ir vietos dimensijoje: kairė, vidurys, dešinė. Taigi gaunami 9 galimi pasirinkimai, kurie yra integruoti į teksto nuspėjimo sistemą.

Dvylikai sveikų savanorių buvo rodoma 3 x 3 simbolių matrica ir per ausines transliuojami devyni, 100 ms trunkantys, garsiniai signalai iššaukiantys vartotojo reakcija. Ši reakcija buvo fiksuojama su elektroencefalogramos kepure.

Teksto nuspėjimui buvo naudojama modifikuota T9 versija. Ji buvo paruošta su 10 000 dažniausiai vartojamų vokiškų žodžių. Taip pat, kadangi standartinė T9 naudoja daugiau nei 9 mygtukus teksto įvedimui sistema buvo modifikuota taip, kad mygtukai nuo 2 iki 9 naudojami teksto įvedimui, o 1 mygtukas perjungia į žodžių pasirinkimo režimą.

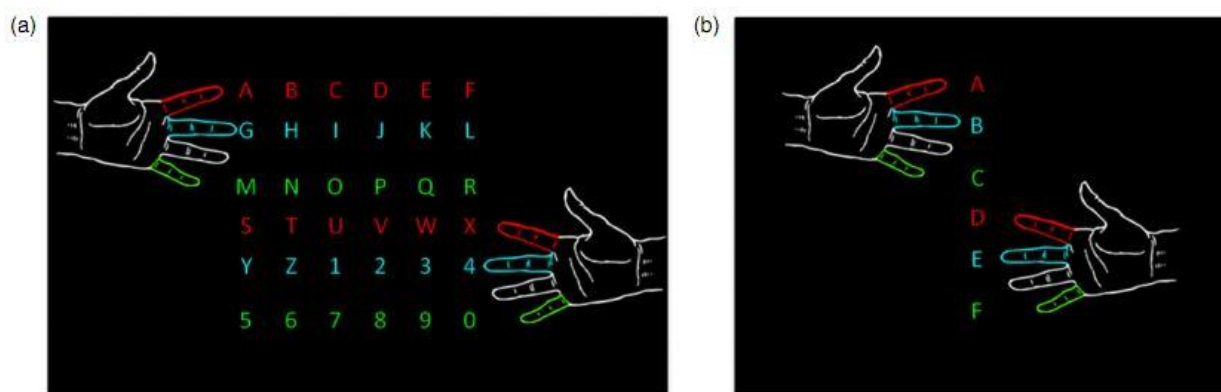
2.4.4. ERP grįsta lytėjimo teksto įvedimo sistema

Marjolein van der Waal ir kolegos iš Radboud University Nijmegen sukūrė lytėjimu valdomą teksto įvedimo sistemą [12]. Vartotojų pirštų galiukai stimuliuojami trumpais intarpais, tuo metu jų smegenų veikla buvo fiksuojama. Raidės buvo priskirtos kiekvienam pirštui ir vartotojas galėjo išsirinkti norimą raidę tyliai skaičiuodamas kiek kartų pirštas su atitinkama raide buvo stimuliuotas.



Pav. 7 Brailio stimulatorius naudotas pirštų galų stimuliuoti

Aukščiau (Pav. 7) matomas prietaisais naudotas sužadinti vartotojo reakciją, o žemiau esančiame paveiksle pavaizduotas raidžių išdėstymas ant pirštų. Tam jog vartotojai nesuspastų stimulatoriaus stipriau viena ranka nei kita buvo naudojami elektrodai pritvirtinti ant kiekvienos rankos alkūnės, kurie fiksavo elektromiogramą.



Pav. 8 Raidžių ir pirštų sąryšis

Paveiksle Pav. 8 pateikiamas raidės išsirinkamas. Jis vyksta dviem žingsniais: visų pirma išsirinkama raidžių grupė (a) ir toliau išrenkama individuali raidė (b).

2.4.5. Rezultatų palyginimas

Šiame poskyryje esančioje lentelėje (Lentelė 1) pateikiami analizuotų programų rezultatų palyginimai.

Lentelė 1 Rezultatų palyginimas

Kriterijus/Programa	DTU BCI	Singapūro IR instituto P300 BCI	2D Garsinė P300	ERP grįsta lytėjimo
Informacijos perdavimo greitis, bitai/min	21.94 ± 15.63	-	5.95	Vidutiniškai 7.8
Simbolių rašymo greitis simboliai/min	4.90 ± 3.84	6 ± 4	Vidutiniškai 0.845	Vidutiniškai 1.51

2.4.6. Programų palyginimas

Žemiau esančioje lentelėje (Lentelė 2) pateikti 2.4 skyriuje apžvelgtų sprendimų duomenys.

Lentelė 2 Programų palyginimas

Kriterijus/Programa	DTU BCI	Singapūro IR instituto P300 BCI	2D Garsinė P300	ERP grįsta lytėjimo
Įrengimas	g.USBamp	Neuroscan NuAmps	Brain Products 64-kanalų stiprintuvas	BioSemi ir pjezoelektrinis Brailio stimulatorius
Signalai	6 – 11 Hz	1 – 24 Hz	100 Hz	35 Hz
Signalų apdorojimas	FFT	Artefaktų pašalinimas, kanalų išrinkimas	MATLAB funkcijos	Downsampling
Naudoti mašininio mokymosi metodai	-	SVM klasifikavimas	Dvejetainis klasifikavimas naudojant Fisherio Diskriminantą	Linijinis klasifikatorius, dvejetainis klasifikatorius
Triukšmo filtravimo metodai	Juostinio praėjimo filtras	Juostinio praėjimo filtras	Juostinio praėjimo filtras	Juostinio praėjimo filtras, judančio vidutinio vagono filtras, Furjė filtras
Antrinis apdorojimas (angl. Post Processing)	-	PCA metodai	-	ANOVA
Sąsaja:				
Simbolių išdėstymas	Matrica 3 x 3	Matrica 4 x 10	Matrica 3x3	Matrica 6 x 6
Ar priklausoma nuo kalbos?	Ne	Ne	Taip (Vokiečių)	Ne
Charakteristikos	SSVE potencialai	P300 potencialai	P300 potencialai ir garsiniai stimulai	P300, N2 potencialai ir Brailio stimulatorius

3. Siūlomas metodas

Šiame skyriuje aprašoma motyvacija kurti konceptais grįstą išmaniąją teksto įvedimo sąsaja bei pats konceptų panaudojimas.

3.1. Įvadas

Tiek Lietuvoje, tiek pasaulyje labai opi problema yra palengvinti neįgaliųjų dalią. Yra tokių ligonių, kurie gali atlikti labai minimalius judesius, tačiau jų nepakanka normaliai funkcionuoti. Todėl jie turi būti slaugomi. Dalis šių neįgaliųjų dėl įvairių priežasčių nesugeba šnekėti ir norėdami komunikuoti su aplinkiniais turi kitais būdais reikšti savo mintis. Šių žmonių galimybės naudotis kompiuteriu yra ribotos, nes kompiuteriai pagrinde vis dar yra valdomi pele ir klaviatūra. Naudojimosi kompiuteriu galimybės minimalus ar visiškai nebuvimas labai suprastina negalią turinčių žmonių gyvenimo kokybę. Todėl tobulėjant kompiuterinėms technologijoms reikia atsižvelgti į žmonių naudojimosi kompiuteriu kokybę, visuose gyvenimo etapuose. Nauji konceptai ir metodai (tokie kaip kasdienį gyvenimą palengvinančios aplinkos (angl. *Ambient Assisted Living*) [14] technologijų įrenginiai, sistemos ir sąsajos) turi būti ištirti, kad sukurti efektyvesnes naudojimosi kompiuteriu sąsajas ne tik tipiniams vartotojams, bet ir neįgaliesiems ar senyvo amžiaus žmonėms. Bendravimo kanalų, tarp vartotojo ir kompiuterio, kiekio padidėjimas leidžia išplėsti neįgalių vartotojų galimybes, padidinant konteksto supratimą ir kompiuterinių sistemų intelektą protingose aplinkose bei padidinant tipinių vartotojų sąveikos lygį, pvz.: kompiuteriniuose žaidimuose.[15]

Nepaisant didelės technologijų augimo spartos dauguma paslaugų yra neprieinamos negalią turintiems vartotojams. Būtent dėl techninės įrangos dizaino trūkumų šie vartotojai negauna prieigos prie informacijos ir žinių, kurias teikia naujausios technologijos.

Neįgalūs individai su daliniais motoriniais sugebėjimais dažniausiai gali naudotis mechaniniais jungikliais bendravimo sistemų valdymui per tiesioginius pasirinkimus [16]. Tačiau individams galintiems judėti tik dalinai ar visai negalintiems judėti („locked in“ sindromas), gali reikėti sąsajų nepasikliaunančių vien paprastais jungikliais. Jungikliai šiose sąsajose galėtų būti aktyvuojami fiksuojant vartotojo mažus judesius infraraudonaisiais spinduliais arba akių mirksniais. Taip pat gali būti pasitelktas elektromiogramos signalų fiksavimas atliekant valingus rankų ar veido raumenų susitraukimus arba okulografiją.

3.2. Konceptų panaudojimas

Piktogramos sutinkamos visur šiandieniniame pasaulyje: medicinoje, kompiuteriuose ir pan. Taip yra todėl, kad jos savo ikoniška forma atvaizduoja realaus ar virtualaus pasaulių objektus, vietas, veiksmus, kryptis, apribojimus ir pan. Šios ikonos yra nepriklausomos nuo kalbos, todėl suteikia kur kas platesnį panaudojimą nei tekstas. Atpažinti paveikslą yra lengviau nei skaityti tekstą. Dėl šių priežasčių ir nuspręsta magistrinio darbe, vietoje konkrečių raidžių ar žodžių, vartotojo-kompiuterio bendravimą patobulinti panaudojant vaizdus, turinčius platesnę prasmę nei vienas žodis – konceptus.

Pavyzdžiui į klausimą „kaip tu jautiesi?“ galima atsakyti rašant žodį „gerai“, tačiau pasitelkus konceptą būtų galima atsakyti teigiamą emociją išreiškiančiu paveikslėliu. Žemiau pateikiamas paveikslėlio pavyzdys (Pav. 9).



Pav. 9 Teigiamą emociją išreiškiantis paveikslukas

Taip pat pasitelkus tą patį konceptą būtų galima pasisveikinti.

Toks konceptų panaudojimas leistų sumažinti atsakymo laiką bei veiksmų reikalingų pateikti atsakymui kiekį. Šis teksto įvedimo sąsajos suabstraktinimas galėtų praplėsti potencialių vartotojų spektrą, nes taip būtų sudarytos galimybės ja naudotis vartotojams, kurie nemoka rašyti ar skaityti ar turi protinių sutrikimų.

3.3. Vizualinės kalbos modeliavimas

Magistro darbe naudojama vizuali kalba – konceptai – sudaryta iš penkių pagrindinių dalių ir vienos į konkrečią vartotojų grupę (negalią turintys vartotojai) orientuotos dalies. Šios dalys tai plačią prasmę turintys konceptai, matomi Pav. 10.



Pav. 10 Pagrindinės vizualinės – konceptų – kalbos dalys

Šie konceptai atitinka tekstinės kalbos dalis. Eilutėmis iš kairės į dešinę:












- Jausmai – atitinka būdvardžius. Šio koncepto vieni vaikų yra „šalta“, „karšta“.
- Veiksmai – atitinka veiksmažodžius. Šio koncepto vieni vaikų – „skaityti“, „bendrauti“.
- Daiktai – atitinka daiktavardžius. Vaikai: „televizorius“, „maistas“.
- Asmuo – daugiaž atitinka įvardžius, nes po šiuo konceptu grupuojami konceptai nurodantys asmenį: „Aš“, „Daktaras“.
- Vieta – šis konceptas apibrėžia vietą nurodančius konceptus, pvz.: „ligoninė“, „namai“.
- Laikas – tai konceptas apjungiantis laiką nusakančius konceptus, pvz.: „šiandien“, „vakar“.
- Kūno dalys – tai bendrinis konceptas orientuotas į tikslinę auditoriją (neįgaliuosius). Šis konceptas apjungia konceptus nurodančius žmogaus kūno dalį, pvz.: „ranka“, „koja“ ir t.t.

- Pagalbiniai konceptai – po šiuo konceptų grupuojami konceptai atliekantis pagalbinės kalbos funkcijas, tokias kaip neigimas, sakinio užbaigimas, klausimas.

Pilnas sistemoje naudojamų konceptų sąrašas pateiktas prieduose (9 skyriuje) esančioje lentelėje – Lentelė 5.

Konceptai ir juos apjungiantys tėviniai konceptai parinkti kuo paprastesni ir universalesni. Tai suteikia vartotojui galimybę bendrais terminais išsakyti savo poreikius. Taip pat tai reiškia, kad įvairius konceptus galima naudoti skirtinguose kontekstuose, suteikiant skirtingas reikšmes sakiniams. Toks konceptų parinkimas ir panaudojimas siūlomai vizualinei kalbai suteikia lankstumo. Žemiau esančioje lentelėje (Lentelė 3) pateikiamos pavyzdinės vizualinės konceptų kalbos frazės.

Lentelė 3 Pavyzdinės vizualinės kalbos frazės

Tekstinė frazė	Vizualinė – konceptų frazė
Aš noriu valgyti	
Kiek dabar valandų?	
Man skaudą širdį, kvieskite greitąją	
Ar tu gali pakasyti man koją?	
Išjunk televizorių	
Man reikia eiti į ligoninę šiandien	
Man reikia pagalbos	
Aš laimingas namie	
Aš noriu eiti į lauką	
Kvieskite gydytoją	
Negaliu kvėpuoti	

4. Projektinė dalis

Šiame skyriuje pateikiami projekto reikalavimai, funkcijos bei sistemos statinis ir dinaminis vaizdai.

4.1. Projekto reikalavimai

4.1.1. Funkciniai reikalavimai

Kuriamai sistemai buvo iškelti tokie funkciniai reikalavimai:

- Sistema turi leisti vartotojui įvesti tekstą pasinaudojant nervų-kompiuterio sąsaja.
- Sistema turi vartotojui suteikti galimybę įvedus dalį žodžio išsirinkti labiausiai tinkantį atitikmenį iš jos nuspėtų variantų.
- Galimybė išsirinkti konceptą.
- Konceptai turi būti sugrupuojami į struktūrinius vienetus, kuriuose gali būti kiti vienetai arba pavieniai konceptai.
- Sistema turi suteikti galimybę vartotojui ištrinti ar pakeisti pasirinktą žodį, simbolį ar konceptą.
- Sistema turi turėti galimybę nuspėti tolimesnę vartotojo įvestį.
- Sistema turi sugebėti automatiškai įterpti nuspėtą žodį ar jo dalį į tekstą.
- Turi būti sukurta galimybė susikurti savo konceptus.
- Turi būti įgyvendintas konceptų šalinimas.

4.1.2. Nefunkciniai reikalavimai

Projekto pradžioje buvo sudarytas nefunkcinių reikalavimų sąrašas:

- Žodyne neturi būti keiksmažodžių ar kitų įžeidžiančių žodžių.
- Sistemoje negali būti įžeidžiančių konceptų.
- Mašininio mokymosi algoritmų atsako greitis turi būti priimtinas vartotojui.
- Konceptų kiekis turi būti pakankamas bent minimaliam bendravimui.

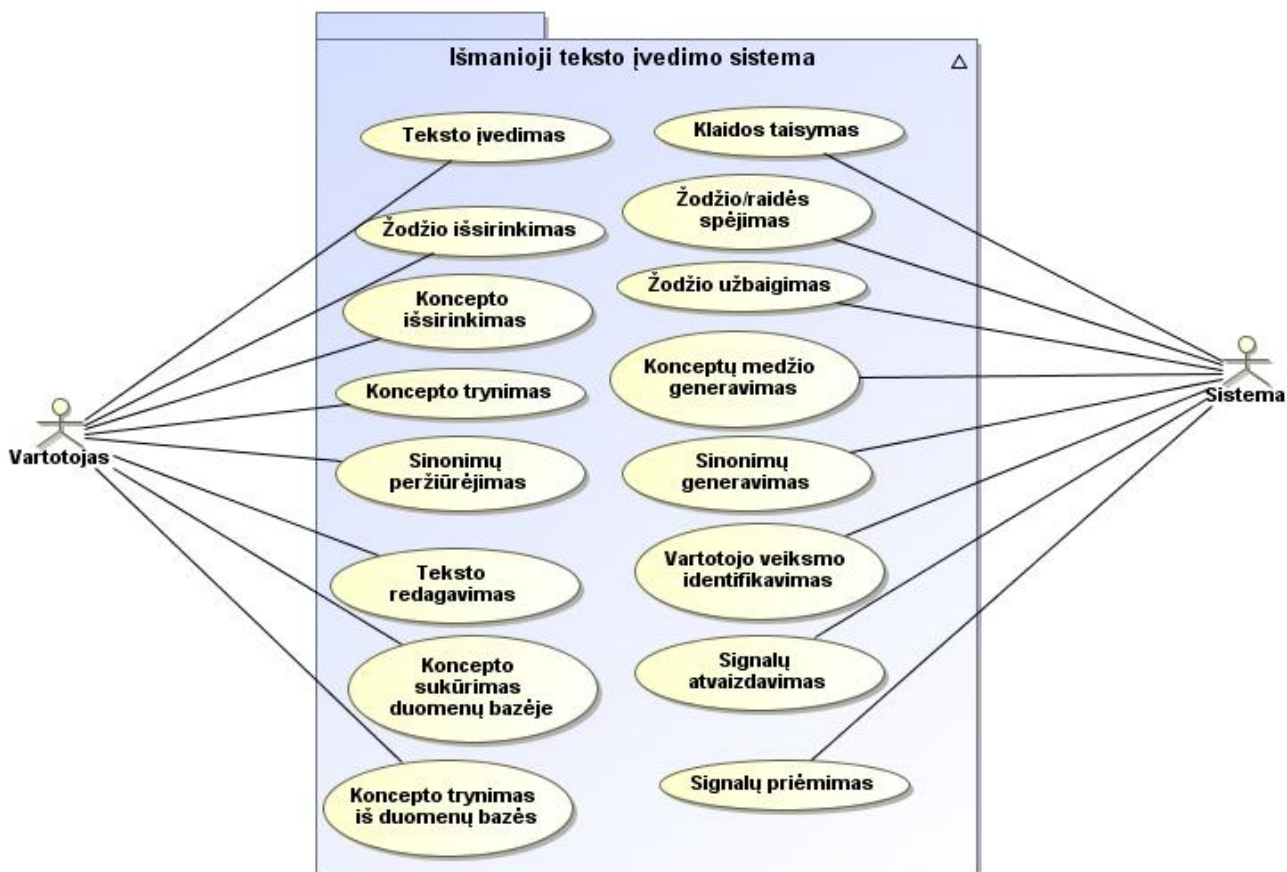
4.2. Programų sistemos funkcijos

Sistemos vartotojas pasinaudodamas NIA prietaisu gali įvesti tekstą. Automatinis žodžių užbaigimas pateikia savo spėjimą, koks žodis yra vedamas. Jei vartotojas pageidauja jis gali pasirinkti šį pasiūlymą. Tada bus automatiškai užbaigiamas vedamas žodis. Ši funkcija tinka ir klaidų taisymui. Jei įvestas pilnas žodis su klaida ir sistema siūlo teisingą variantą, galima jį išsirinkti ir taip ištaisyti klaidą.

Kitas svarbus projekto aspektas yra konceptai. Vartotojas gali įvesti arba trinti jau parašytus konceptus. Sistema sukonstruoja konceptų medį, tačiau vartotojas gali jį koreguoti savo nuožiūra. Yra sukurtos galimybės pridėti savo konceptus į duomenų bazę arba šalinti iš jos jau egzistuojančius.

Visi konceptai yra susieti su juos atitinkančiais žodžiais – sinonimais, todėl galima peržiūrėti konceptais parašyto teksto atitikmenį žodžiais.

Žemiau esančiame paveiksle (Pav. 11) matoma sistemos panaudos atvejų diagrama.



Pav. 11 Panaudos atvejų diagrama

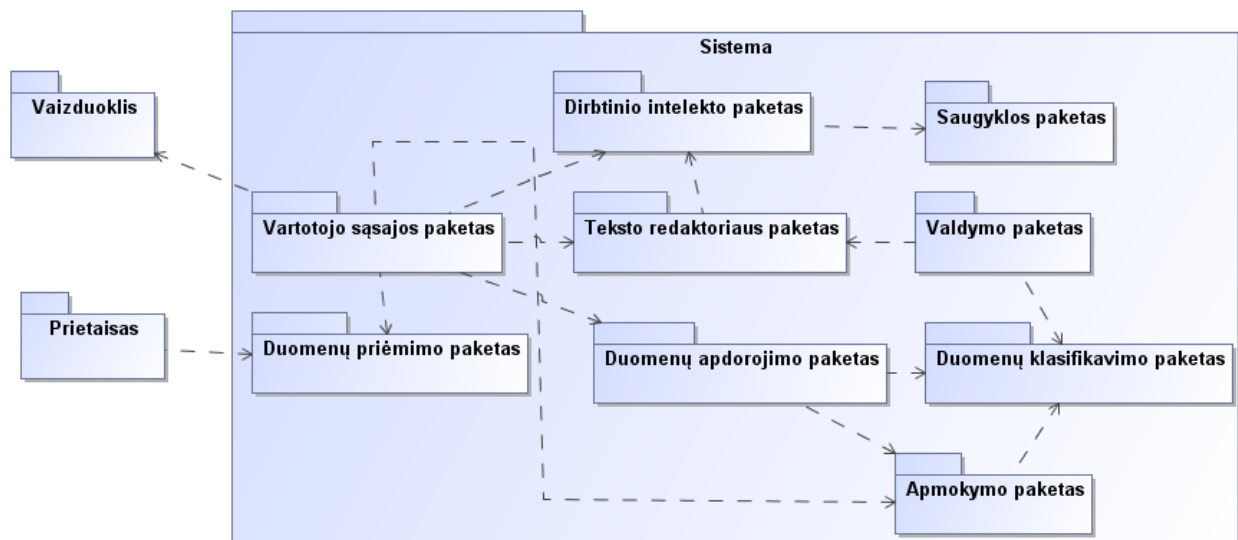
4.3. Architektūros tikslai

Kuriant sistemos architektūrą buvo užsibrėžti tokie tikslai:

- Sistema turi būti projektuojama su galimybe ateityje ją išplėsti.
- Turi būti panaudoti mašininio mokymosi algoritmai teksto įvedimui pagreitinti.
- Tam, kad sistemą padaryti labiau prieinamą įvairiems vartotojų tipams, turi būti naudojami konceptai.
- Turi būti sudarytos galimybės plėsti žodyną ir konceptų sąrašą.
- Vartotojui turi būti sudarytos sąlygos atšaukti atliktą komandą.
- Sistema turi turėti grafinę sąsają.

4.4. Sistemos statinis vaizdas

Žemiau esančioje diagramoje (Pav. 12) pateikiamas sistemos paketų išdėstymo vaizdas.



Pav. 12 Sistemos išdėstymo diagrama

Vartotojo sąsajos paketas – šis paketas atsakingas už grafinės vartotojo sąsajos langų atvaizdavimą ir veikimą.

Teksto redaktoriaus paketas – paketo pagalba bus įvedinėjamas tekstas.

Valdymo paketas – atsakingas už valdymo funkcijų įgyvendinimą.

Dirbtinio intelekto paketas – šis paketas įgyvendins mašininio mokymosi algoritmų funkcijas.

Saugyklos paketas – suteiks prieigą prie žodžių ir konceptų duomenų bazės.

Duomenų priėmimo paketas – priims duomenis iš prietaiso.

Duomenų apdorojimo paketas – šis paketas įgyvendins gautų duomenų apdorojimą.

Apmokymo paketas – paketas atsakingas už klasifikavimo modelio sukūrimą.

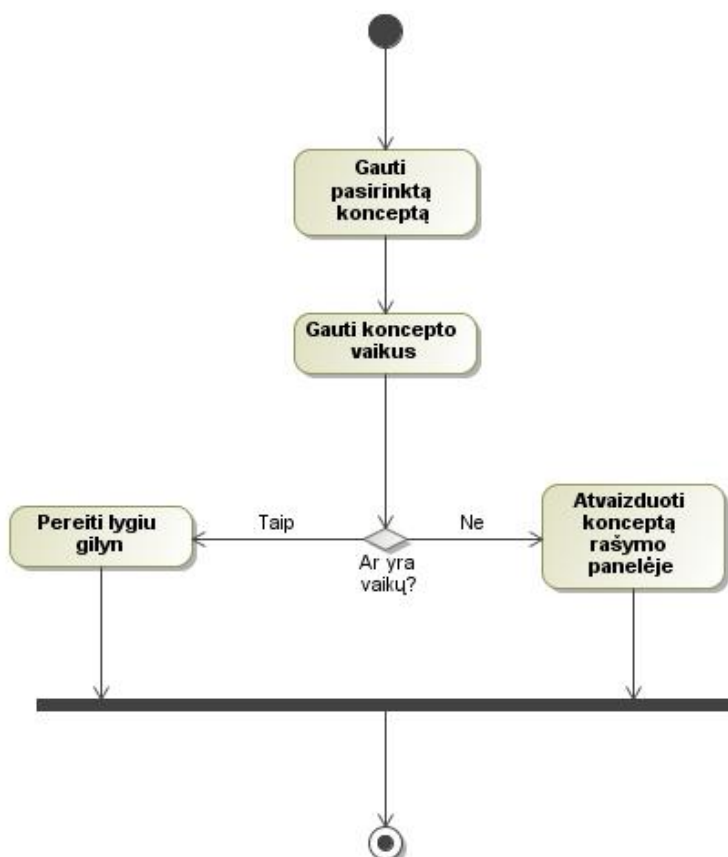
Duomenų klasifikavimo paketas – pasinaudodamas klasifikavimo modeliu sujungs protines būsenas su valdymo komandomis.

4.5. Sistemos dinaminis vaizdas

Šiame poskyryje pateikiami kelių pagrindinių sistemos atliekamų veiksmų aprašymai bei veiklos diagramos.

4.5.1. Koncepto išsirinkimas

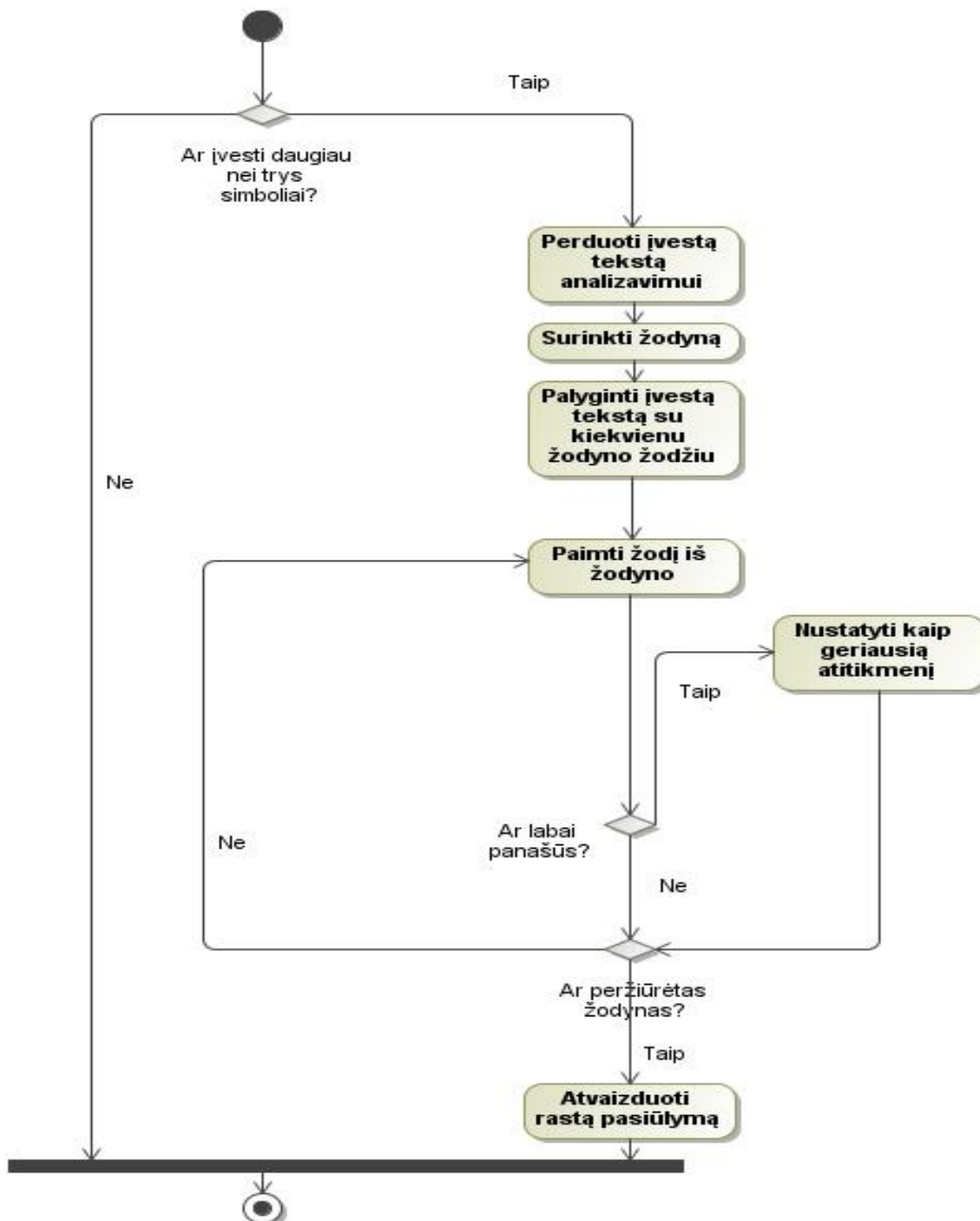
Žemiau pateiktoje diagramoje (Pav. 13) matoma koncepto išsirinkimo diagrama. Ši veiksmų seka pradedama kai identifikuojamas koncepto pasirinkimo veiksmas. Visų pirma paimamas pasirinktas konceptas, tada patikrinama ar jis turi vaikų. Jei egzistuoja pasirinkto koncepto vaikai, tai reiškia, jog galima pereiti lygiu žemiau prie konkretesnių konceptų – vaikų. Tada atliekamas šis veiksmas. Jeigu vaikinių konceptų nėra, tada į rašymo panelę įtraukiamas pasirinktas konceptas.



Pav. 13 Koncepto išsirinkimo veiklos diagrama

4.5.2. Pasiūlymo generavimas ir atvaizdavimas

Pasiūlymo generavimas inicijuojamas kai vartotojas įveda daugiau nei 3 simbolius. Įvestas tekstas yra palyginamas su visais žodyne esančiais žodžiais. Taip išrenkamas panašiausias žodis, kuris ir yra pasiūlomas vartotojui. Šis veiksmas kartojamas su kiekviena naujai įvesta raide. Žemiau esančiame paveiksle (Pav. 14) pateikta pasiūlymo generavimo ir atvaizdavimo diagrama.



Pav. 14 Pasiūlymo generavimo ir atvaizdavimo diagrama

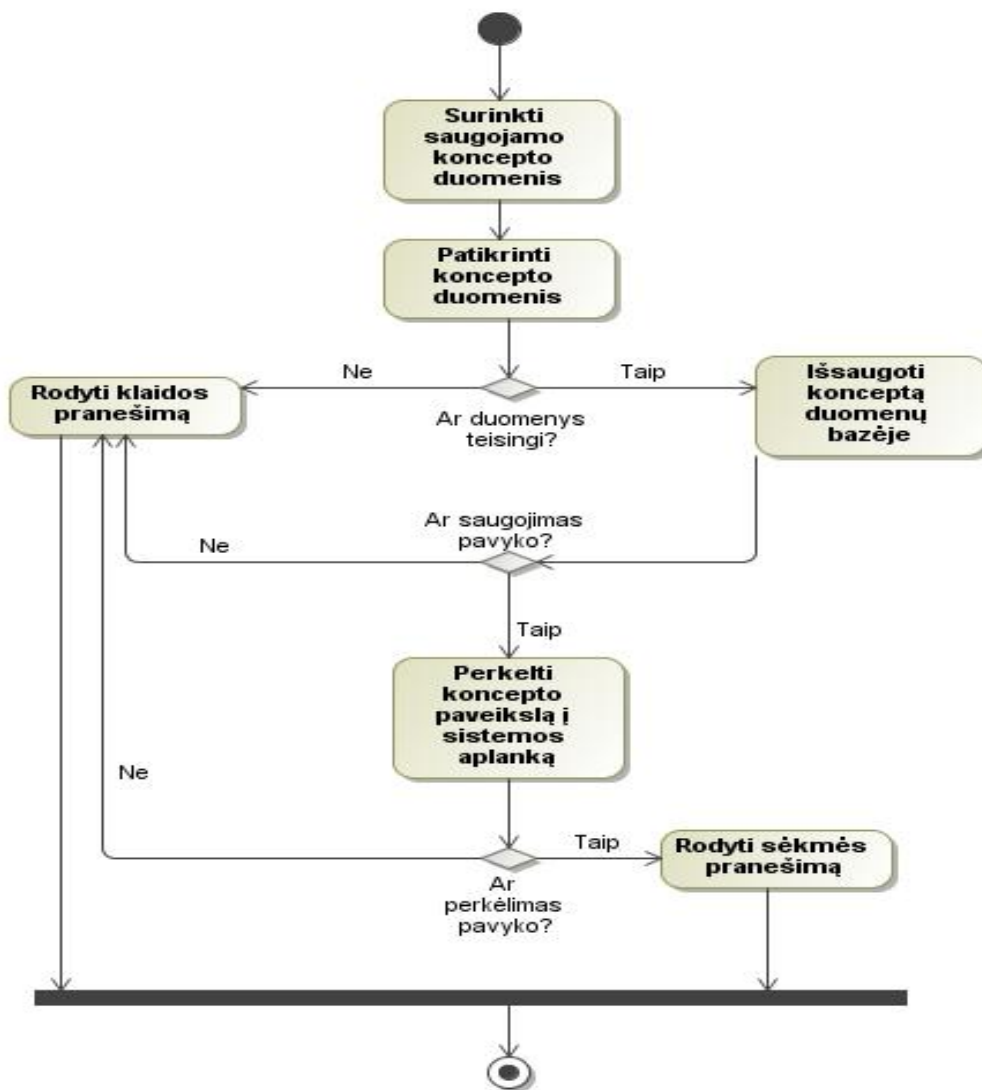
4.5.3. Naujo koncepto saugojimas

Šis veiksmas vykdomas suvedus koncepto duomenis ir nuspaudus saugojimo mygtuką. Visų pirma patikrinama ar įvestas koncepto pavadinimas ir ar pasirinktas paveikslėlis yra 64 x 64 pikselių didumo bei ar jo plėtinys yra „png“. Nepavykus patikrinimui parodomas nesėkmės pranešimas su žinute apie konkrečia bloga vieta.

Sėkmingai atlikus duomenų tikrinimą konceptas yra įrašomas į duomenų bazę ir po to pasirinktas paveikslas yra perkeliamas į sistemos aplanką. Jei atliekant kurį nors iš šių veiksmų susiduriama su klaida visi veiksmai yra atšaukiami ir vartotojui parodomas klaidos pranešimas.

Baigus saugojimą sėkmingai vartotojui pateikiamas pranešimas apie sėkmingą išsaugojimą.

Paveiksle (Pav. 15) pateikta koncepto saugojimo veiklos diagrama.



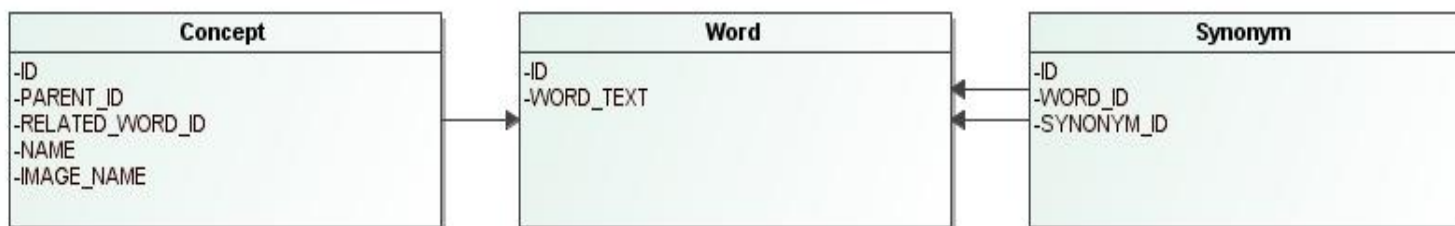
Pav. 15 Naujo koncepto saugojimo veiklos diagrama

4.6. Duomenų vaizdas

Paveiksle Pav. 16 matoma duomenų bazės esybių diagrama. Pagrindinė lenta yra „*Concept*“, kurioje saugoma koncepto informacija:

- ID – identifikatorius.
- PARENT_ID – tėvinio koncepto identifikatorius.
- RELATED_WORD_ID – susijusio žodžio identifikatorius.
- NAME – pavadinimas.
- IMAGE_NAME – paveikslėlio pavadinimas.

Lentoje „*Word*“ saugomas žodžio identifikatorius bei pats žodis. „*Synonym*“ lenta skirta susieti žodžius kaip sinonimus. Toks susiejimas leidžia išvengti duomenų dubliavimo, nes egzistuoja tik viena lenta laikanti žodžius. Sinonimų lentoje saugomas identifikatorius, žodžio bei jo sinonimo identifikatoriai.



Pav. 16 Duomenų bazės esybių diagrama

4.7. Testavimas

Sistemos komponentų pavienės funkcijos buvo ištestuotos vienėtų testais, o grafinė vartotojo sąsaja ir bendras sistemos veikimas testuotas integraciniais testais.

Komponentai testuoti vienėtų testais:

1. Saugyklos paketas.
2. Vartotojo sąsajos paketas.
3. Dirbtinio intelekto paketas.

Bendram sistemos funkcionalumui patikrinti buvo naudojami integraciniai testai, kurie apjungė tokius sistemos paketus:

4. Saugyklos paketas.
5. Vartotojo sąsajos paketas.
6. Dirbtinio intelekto paketas.
7. Teksto redaktoriaus paketas.
8. Valdymo paketas.

Integravimo testavimo metu tikrinta kaip sistemos dalys bendradarbiauja: ar duomenų nuskaitymo modulis gauna ir persiunčia duomenis iš prietaiso. Kaip sąveikauja duomenų apdorojimo, klasifikavimo, ir apmokymo paketai. Ar valdymo paketas iškviečia reikiamas teksto redaktoriaus funkcijas. Į valdymo paketą buvo paduodamos užklauskos simuliuojančios vienos ar kitos valdymo komandos inicijavimą. Galiausiai surinkus vartotojo sąsajos objektus jie buvo palyginami su lauktu rezultatu.

Tam, kad izoliuoti testuojamus modulius nuo kitų testavimo metu šalutinių moduliai buvo pakeisti netikrais (angl. mock) objektais, kurie leido gražinti pageidautiną būseną, bei stebėti į juos ateinančius kvietimus ir duomenis.

5. Eksperimentinė dalis

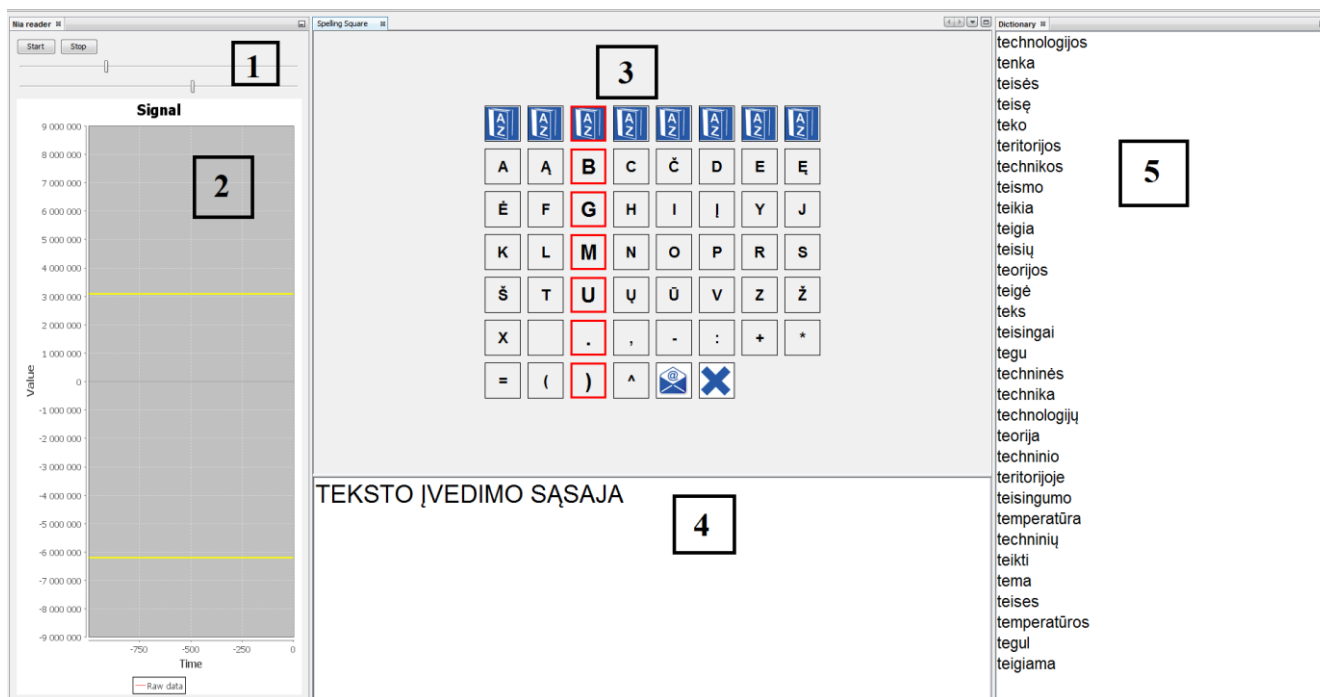
Šiame skyriuje aprašomas atliktas konceptais grįstos išmaniosios teksto įvedimo sistemos tyrimas lyginant rašymo greitį su tradicine teksto įvedimo sąsaja. Taip pat pateikiami tyrimo rezultatai.

5.1. Tyrime naudotų teksto įvedimo sistemų aprašymas

Sistema, kurioje sukurta konceptais grįsta teksto įvedimo sąsaja yra ir tradicinė teksto įvedimo sąsaja. Šiame poskyryje pateikiami abiejų tyrime naudotų sąsajų aprašymai.

5.1.1. Tradicinė teksto įvedimo sistema

Sistema turi dvi teksto įvedimo sąsajas, viena iš jų yra tradicinė simbolių matrica grįsta teksto įvedimo sąsaja. Šios sąsajos grafinis vaizdas matomas žemiau esančiame paveiksle Pav. 17.



Pav. 17 Tradicinės teksto įvedimo sąsajos vaizdas

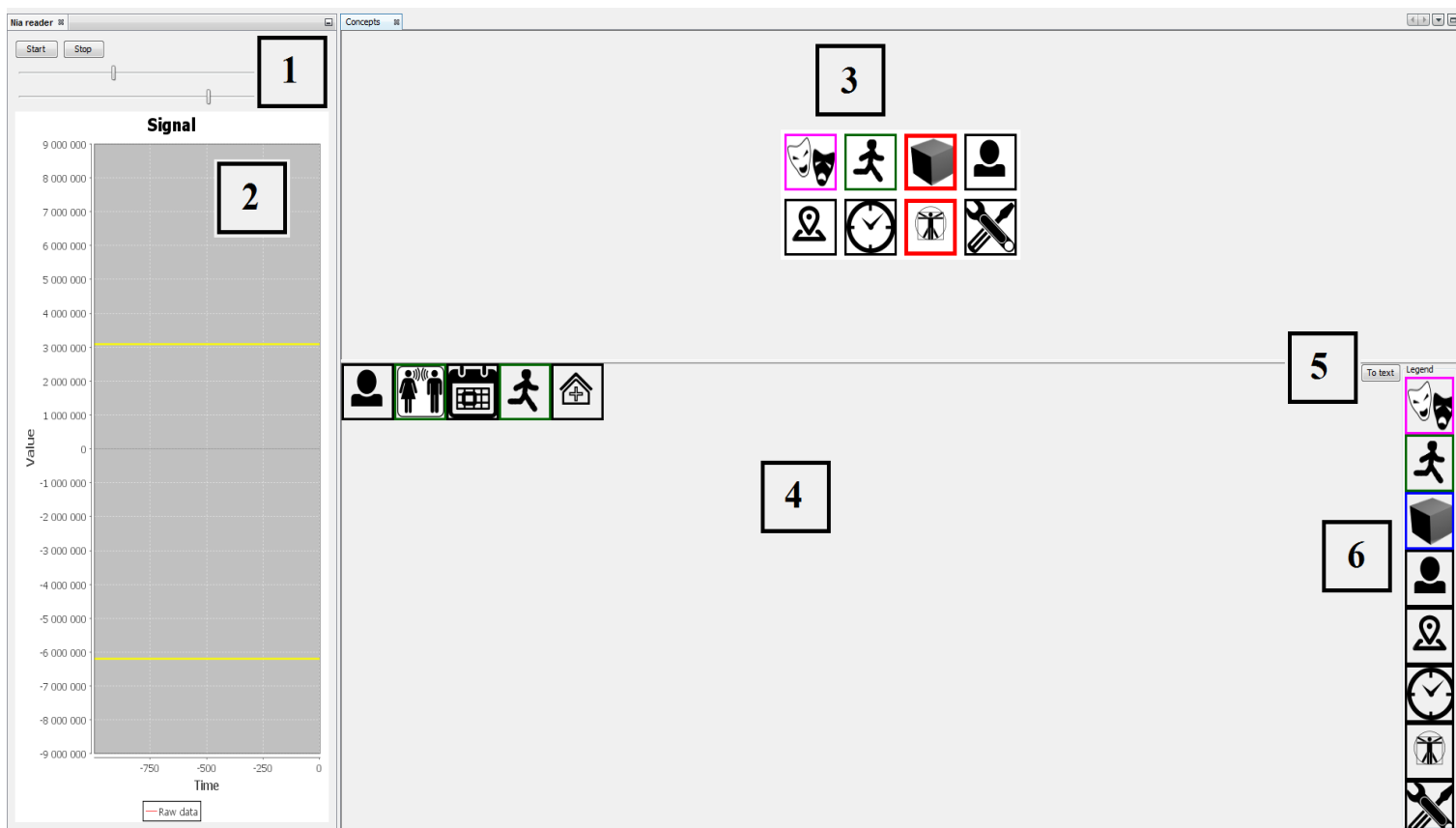
Pirmu numeriu (1) pažymėta signalo ribų reguliavimo slankikliai. Jais reguliuojamos ribos ties kuriomis fiksuojama įvedimo komanda. Šios ribos matomos antru numeriu (2) pažymėtoje signalų atvaizdavimo panelėje. Svarbiausia šios sąsajos dalis yra trečiu numeriu (3) pažymėta simbolių matrica. Iš šios matricos pasirinktas simbolis įrašomas į ketvirtu numeriu (4) pažymėta teksto atvaizdavimo panele. Raudonai pažymėtas stulpelis yra dabartinio pasirinkimo stulpelis. Išsirinkus konkretų stulpelį pradedama iteruoti per to stulpelio simbolius – pažymint kiekvieną individualų simbolį raudonai. Pasirinkimo žymėjimas juda greičiu kintančiu nuo 500 iki 1500 milisekundžių. Šis greitis kinta priklausomai nuo klaidų, kurias atlieka vartotojas įvedinėdamas žodį, skaičiaus. Klaida laikomas įvesto simbolio ištrynimasis arba

pasirinkimo atšaukimas. Kuo mažiau klaidų tuo greičiau juda pasirinkimas. Tai leidžia patyrusiam vartotojui greičiau įvedinėti tekstą.

Viršutinėje simbolių matricos eilutėje išdėstytos žodyno ikonos. Pasirinkus bet kurią iš šių ikonų pereinama į penktu numeriu (5) pažymėtą žodyno panelę. Šioje panelėje iteruojama per labiausiai tinkamus žodžius, išrinktus pagal paskutinę vartotojo įvestį. Pasirinktas norimas žodis įterpiamas į teksto panelę vietoje pradėto vesti žodžio.

5.1.2. Konceptais grįsta teksto įvedimo sistema

Kita sistemos sąsaja yra grįsta konceptais. Šios sąsajos vartotojo sąsajos vaizdas matomas paveiksle Pav. 18.



Pav. 18 Konceptais grįstos teksto įvedimo sąsajos vaizdas

Kaip ir tradicinėje simbolių matrica grįsta teksto įvedimo sistemoje pirmu numeriu (1) pažymėti veiksnių fiksavimo ribų slankikliai. Antru numeriu (2) pažymėta signalo atvaizdavimo panelė. Trečiu numeriu (3) pažymėta panelė, kurioje yra konceptų matrica. Pasirinkimas vykdomas identišškai tradicinei teksto įvedimo sąsajai: pirma išsirenkamas stulpelis, tada pasirenkamas konceptas. Jei konceptas turi vaikinių konceptų, tada pereinama lygiu gilyn, kur vėl vykdomas pasirinkimas. Grįžti į šakninį lygį galima su paveiksle Pav. 19 pavaizduotu konceptu.



Pav. 19 Grįžimo į šakninį lygį konceptas

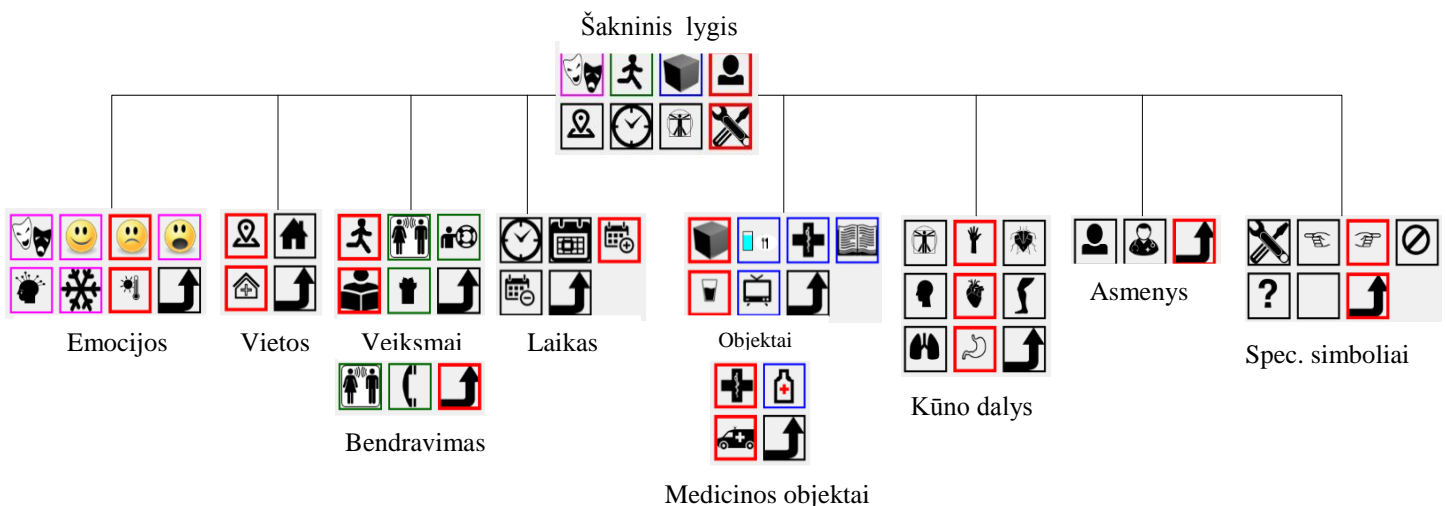
Išsirinkus norimą konceptą jis įtraukiamas į ketvirtu numeriu (4) pažymėtą rašymo panelę. Iš kur jis gali būti ištrintas įvykdžius atšaukimo komandą.

Penktuoju numeriu (5) pažymėtoje vietoje yra teksto generavimo mygtukas. Nuspaudus šį mygtuką pateikiamas langas su konceptus atitinkančiais žodžiais.

Šeštuoju numeriu (6) pažymėta konceptų legenda – joje pateikiami šakninio lygio konceptai apvesti atitinkamomis spalvomis. Pagal šias spalvas galima nustatyti koks konceptas priklauso kuriam šakniniam konceptui.

Didžioji dalis konceptų ikonų paimtos iš „The Noun Project“ (<http://thenounproject.com/>) svetainės, o likusios sukurtos pasinaudojant piešimo programa GIMP.

Kiekvienas konceptas gali turėti vaikinių konceptų. Naujai kuriamas konceptas priskiriamas tam tikram tėviniam konceptui. Taip galima nesunkiai plėsti konceptų medį. Medžio gylis nėra ribojamas todėl galima įtraukti vis konkretesnius konceptus. Tyrime naudota konceptų medžio struktūra pateikta žemiau esančiame paveiksle Pav. 20.



Pav. 20 Konceptų medžio struktūra

5.2. Eksperimento rezultatai

Šiame poskyryje pateikiami su abejomis sąsajomis atliktų eksperimentų aprašymai bei rezultatai.

5.2.1. Eksperimentas su tradicine teksto įvedimo sąsaja

Eksperimentas buvo atliktas su 5 asmenimis iš kurių trys buvo vyrai. Dalyvių amžius nuo 24 iki 54 metų (vidurkis 33 m.). Eksperimente dalyvavę asmenys neturi jokių neurologinių sveikatos sutrikimų, nevartoja vaistų ir turi normalią arba pakoreguotą (nešioja akinius) regą. Dalyviai davė sutikimą atlikti eksperimentą.

Fiziologiniai signalai (elektromiograma) iš subjektų buvo surenkami OCZ NIA aparatu. Vartotojo sąsaja buvo pateikta 13.3 colių įstrižainės TFT LCD ekrane, kurio rezoliucija 1360 * 768 pikseliai ir atnaujinimo dažnis – 60 Hz. Subjektai buvo pasodinti prie stalo. Kompiuterio ekranas buvo stalo viduryje ir

apytiksliai 100 cm atstumu nuo subjekto. Kiekvieno simbolio dydis buvo 1.5 * 1.5 cm ir visa teksto įvedimo matrica buvo 9.5 * 13 cm dydžio.

Eksperimento dalyviams buvo duotos įvesti trys teksto pastraipos. Dalyviai turėjo suvesti šias pastraipas pasinaudodami teksto įvedimo sąsaja. Visas tekstas buvo pateiktas lietuvių kalba. Pirma pastraipa sudaryta iš 126 simbolių ir atitinka kasdieniame bendravime naudojamą kalbą. Antroje pastraipoje yra 111 simbolių. Šioje pastraipoje pateikiamas mokslinis tekstas. Trečiojoje pastraipoje yra 120 simbolių, o joje esantis tekstas yra mokslinis su matematinėmis lygtimis. Kiekvienas eksperimentas buvo kartotas 4 kartus. Eksperimentų metu buvo fiksuojamas vidutinis tikslumas, įvedimo greitis ir bitų perdavimo sparta.

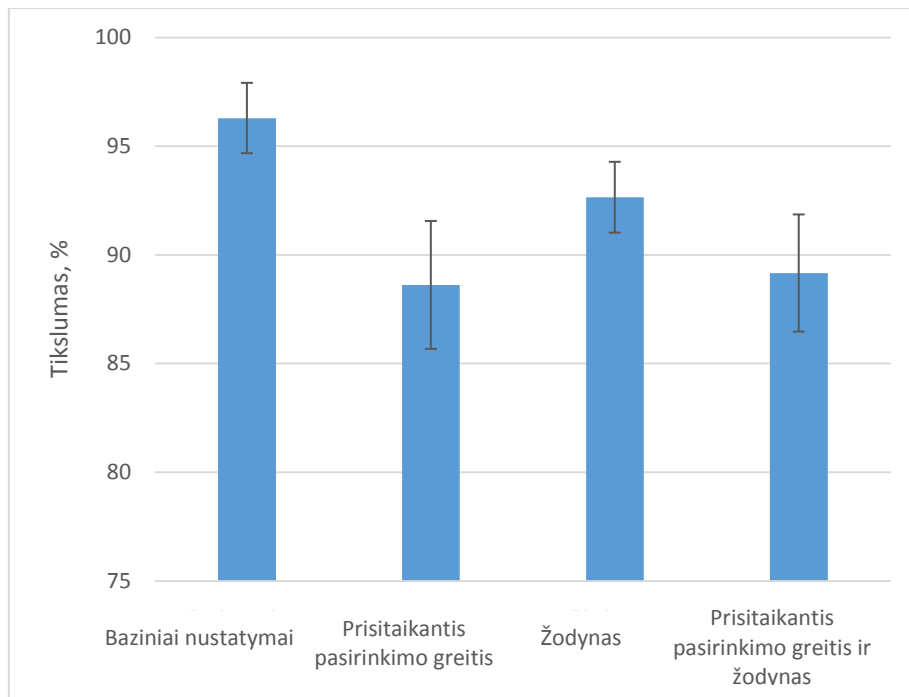
Teksto įvedimo sąsaja gali būti įvertinta apskaičiuojant įvedimo tikslumą, bitų perdavimo bei įvedimo greičio metrikas. Įvedimo tikslumo procentas paskaičiuojamas įvertinant tiksliai atliktų ir iš viso atliktų komandų kiekius.

Bitų perdavimo greitis apibūdina kiek informacijos perduodama per laiko periodą. Įvedimo greičio metrika parodo vidutinį laiką, kurio reikia norint įvesti tekstą. Eksperimento rezultatai pateikti žemiau esančioje lentelėje (Lentelė 4).

Lentelė 4 Tradicinės teksto įvedimo sąsajos eksperimento rezultatai

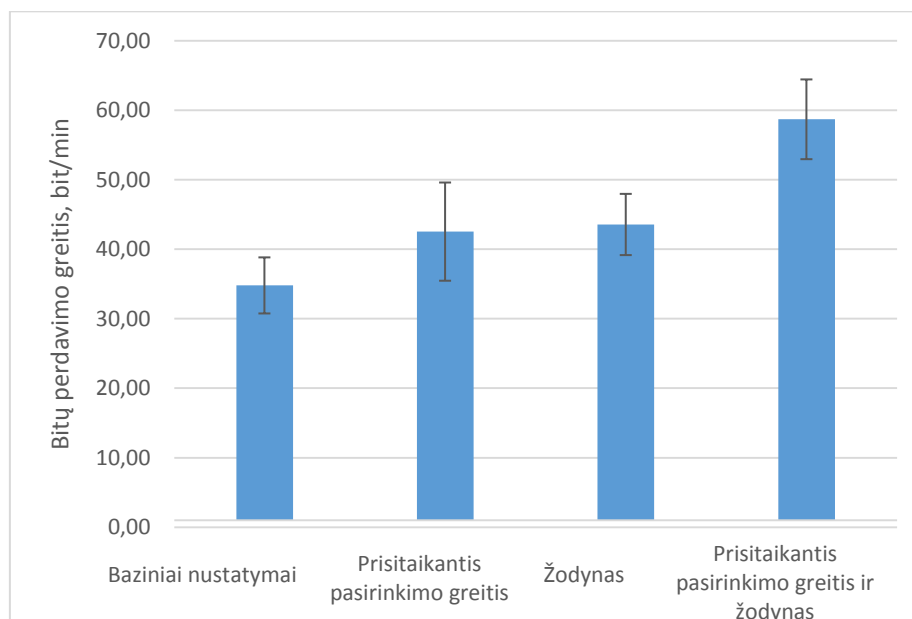
Metrika	Vidutinė reikšmė	Aukščiausia reikšmė
BAZINIAI NUSTATYMAI		
Tikslumas	96.29	98.25
Bitų perdavimo sparta	34.78	41.83
Įvedimo greitis	6.37	7.57
PRISITAIKANTIS PASIRINKIMO GREITIS		
Tikslumas	88.61	93.64
Bitų perdavimo sparta	42.53	49.79
Įvedimo greitis	8.19	9.60
SU ŽODYNU		
Tikslumas	92.65	96.06
Bitų perdavimo sparta	43.55	49.26
Įvedimo greitis	8.22	9.35
SU PRISITAIKANČIU PASIRINKIMO GREIČIU IR ŽODYNU		
Tikslumas	89.16	92.53
Bitų perdavimo sparta	58.69	65.53
Įvedimo greitis	11.35	12.42

Žemiau esančiame paveiksle (Pav. 21) pateikiamas eksperimento metu apskaičiuotos vidutinio tikslumo metrikos grafikas.



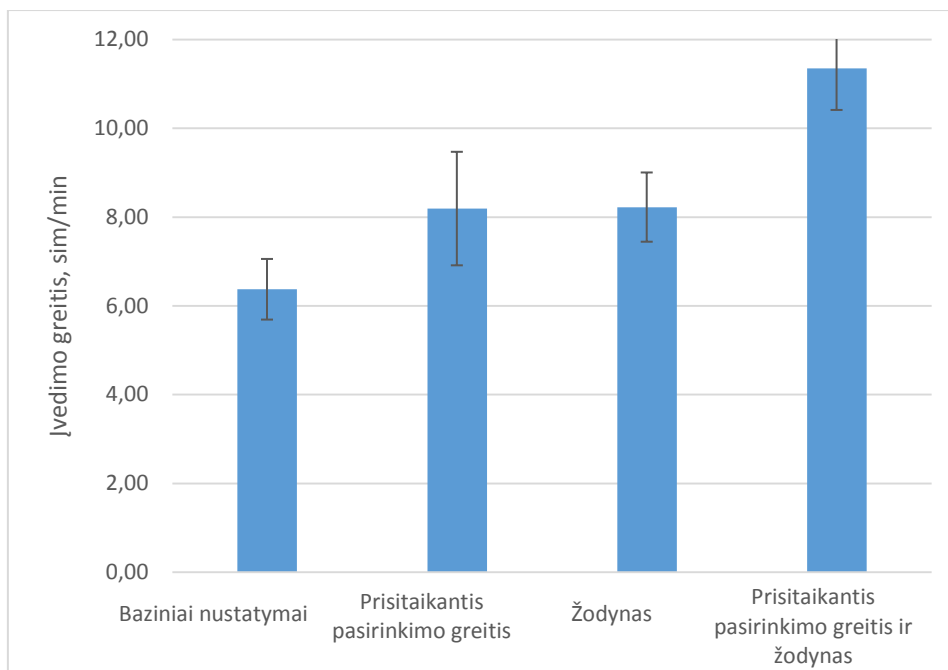
Pav. 21 Tradicinės sąsajos tikslumas

Paskaičiuotas bitų perdavimo greitis matomas Pav. 22 paveiksle.



Pav. 22 Tradicinės sąsajos bitų perdavimo greitis

Paveiksle, Pav. 23, matomi įvedimo greičio įvertinimai prie atitinkamų teksto įvedimo sąsajos nustatymų.



Pav. 23 Tradicinės sąsajos įvedimo greitis

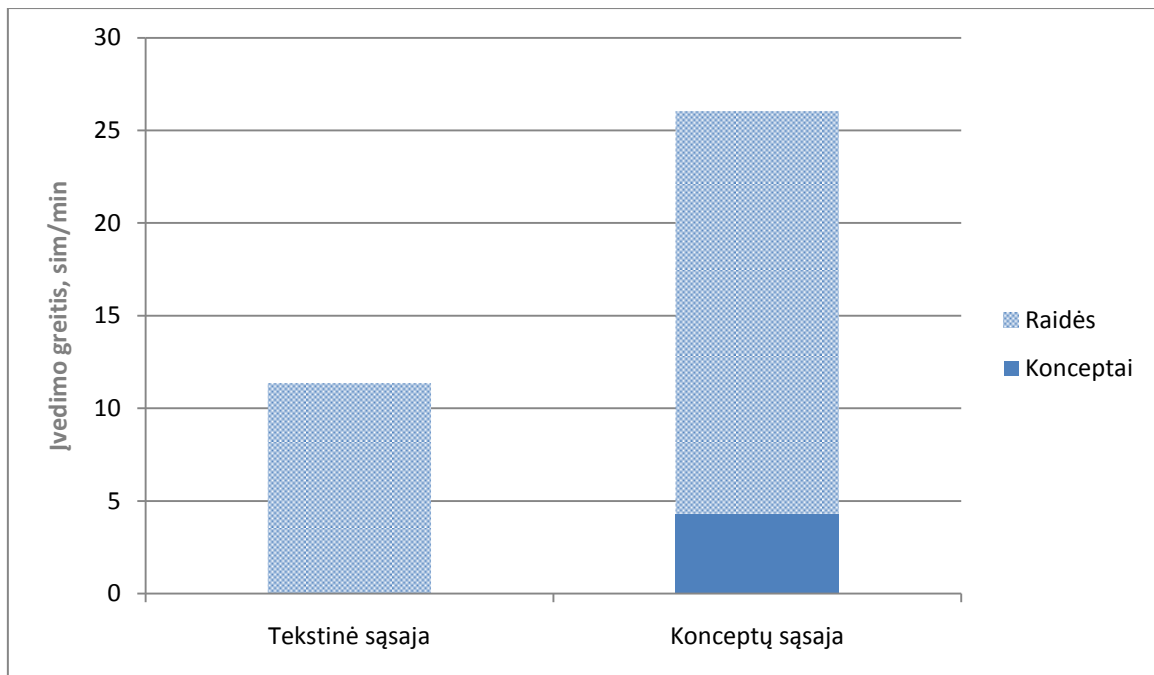
5.2.2. Eksperimentas su konceptais grįsta teksto įvedimo sąsaja

Eksperimentas su konceptais grįsta teksto įvedimo sąsaja buvo atliktas tomis pačiomis aplinkybėmis kaip ir su tradicine teksto įvedimo sąsaja. Eksperimente dalyvavo du vyriškosios lyties subjektai, kurių amžius 24 m. ir 28 m. atitinkamai. Dalyviams buvo pateiktas tekstas sudarytas iš 196 simbolių. Tekstas buvo sudarytas toks, kad atspindėtų kasdieninėje vartotojo kalboje pasitaikančias pagrindinių poreikių formuluotes. Paveiksle Pav. 24 pateikta galima konceptų formuluotė. Šis pavyzdys reiškia „Aš nenoriu valgyti“.



Pav. 24 Konceptų formuluotės pavyzdys

Eksperimento metu buvo registruojamas įvedimo greitis. Žemiau esančiame paveiksle (Pav. 25) pateiktas konceptais grįstos teksto įvedimo sąsajos įvedimo greitis.



Pav. 25 Konceptais grįstos sąsajos įvedimo greitis

Šioje diagramoje (Pav. 25) pateikiamos dvi metrikos: konceptų įvedimo greitis (teisingai įvestų konceptų kiekis per laiko periodą) ir įvedimo greitis simbolių, kurias reikėtų įvesti per laiko periodą norint perteikti tą pačią žinutę, kaip ir konceptais. Iš diagramos matome, kad konceptų įvedimo greitis - 4,31 konceptai/min atitinka 26,03 sim/min. Naudojantis tradicine simbolių matrica grįsta teksto įvedimo sąsaja buvo pasiektas tik 11,35 sim/min įvedimo greitis (Pav. 23 Tradicinės sąsajos įvedimo greitis). Taigi matome virš dviejų kartų didesnę įvedimo greitį konceptais grįstoje teksto įvedimo sąsajoje.

6. Išvados

1. Atlikta problemos sprendimo pasaulyje analizė leido nustatyti panašių teksto įvedimo sąsajų įvertinimo metrikas: informacijos perdavimo greitis bei simbolių įvedimo greitis. Šios metrikos panaudotos vertinant sukurtą konceptais grįstą teksto įvedimo sąsają.
2. Sukurta vizualinė kalba – konceptai, leidžianti vartotojui perteikti savo mintis ikonomis. Konceptai sugrupuoti į logines grupes (Emocijos, daiktai, kūno dalys), kurios suteikia vartotojui iš paprastų konceptų sudaryti sudėtingas frazes. Frazijų pavyzdžiai pateikti – Lentelė 3 Pavyzdinės vizualinės kalbos frazės.
3. Konceptais grįsta teksto įvedimo sąsaja suteikia galimybę atsiriboti nuo kalbos, nes ikonos yra kuriamos, kad būtų atpažįstamos daugumos žmonių.
4. Atliktas eksperimentas su tradicine simbolių matrica grįsta teksto įvedimo sąsaja bei konceptais grįsta teksto įvedimo sąsaja.
5. Abi sąsajos palygintos pagal simbolių įvedimo greičio metriką, nes buvo tiriama kaip konceptai įtakoja teksto įvedimo greitį.
6. Pastebėta, kad konceptai leido pasiekti virš dviejų kartų (2,3) didesnę įvedimo greitį, lyginant su simbolių matrica grįsta teksto įvedimo sąsaja. Taip yra todėl, nes konceptai gali perteikti platesnę prasmę nei konkretus žodis. Pavyzdžiui konceptas reprezentuojantis objektą – laikrodį papildytas konceptu reprezentuojančiu klausuko simboliu, atitiks sakinį „Kiek dabar valandų?“. Taip dviem konceptais reprezentuojamas 19 simbolių tekstas.

7. Literatūra

- [1] Changmok Choi; Hyonyoung Han; Chunwoo Kim; Jung Kim, "Development and Performance Evaluation of a Neural Signal-based Assistive Computer Interface," Robot and Human interactive Communication, 2007. RO-MAN 2007. The 16th IEEE International Symposium on , vol., no., pp.944,949, 26-29 Aug. 2007
- [2] Huiqi Li; Yuanqing Li; Cuntai Guan, "An Effective BCI Speller Based on Semi-supervised Learning," Engineering in Medicine and Biology Society, 2006. EMBS '06. 28th Annual International Conference of the IEEE , vol., no., pp.1161,1164, Aug. 30 2006-Sept. 3 2006
- [3] Thulasidas, M.; Cuntai Guan, "Optimization of BCI Speller Based on P300 Potential," Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the , vol., no., pp.5396,5399, 2005

- [4] Hohne, J.; Schreuder, M.; Blankertz, B.; Tangermann, M., "Two-dimensional auditory p300 speller with predictive text system," Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE , vol., no., pp.4185,4188, Aug. 31 2010-Sept. 4 2010
- [5] Schalk, G.; McFarland, D.J.; Hinterberger, T.; Birbaumer, N.; Wolpaw, J.R., "BCI2000: a general-purpose brain-computer interface (BCI) system," Biomedical Engineering, IEEE Transactions on , vol.51, no.6, pp.1034,1043, June 2004
- [6] Tianyou Yu; Yuanqing Li; Jinyi Long; Cong Wang, "A brain-computer interface controlled mail client," Engineering in Medicine and Biology Society (EMBC), 2013 35th Annual International Conference of the IEEE , vol., no., pp.2164,2167, 3-7 July 2013
- [7] Yuanqing Li; Jiahui Pan; Fei Wang; Zhuliang Yu, "A Hybrid BCI System Combining P300 and SSVEP and Its Application to Wheelchair Control," Biomedical Engineering, IEEE Transactions on , vol.60, no.11, pp.3156,3166, Nov. 2013
- [8] Leeb, R.; Sagha, H.; Chavarriaga, R.; del R Millan, J., "Multimodal Fusion of Muscle and Brain Signals for a Hybrid-BCI," Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE , vol., no., pp.4343,4346, Aug. 31 2010-Sept. 4 2010
- [9] Chaunchu Wang; Cuntai Guan; Haihong Zhang, "P300 Brain-Computer Interface Design for Communication and Control Applications," Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the , vol., no., pp.5400,5403, 17-18 Jan. 2006
- [10] Vilic, A.; Kjaer, T.W.; Thomsen, C.E.; Puthusserypady, S.; Sorensen, H.B.D., "DTU BCI speller: An SSVEP-based spelling system with dictionary support," Engineering in Medicine and Biology Society (EMBC), 2013 35th Annual International Conference of the IEEE , vol., no., pp.2212,2215, 3-7 July 2013
- [11] Han-Jeong Hwang, Jeong-Hwan Lim, Young-Jin Jung, Han Choi, Sang Woo Lee, Chang-Hwan Im, Development of an SSVEP-based BCI spelling system adopting a QWERTY-style LED keyboard, Journal of Neuroscience Methods, Volume 208, Issue 1, 30 June 2012, Pages 59-65, ISSN 0165-0270
- [12] Introducing the tactile speller: an ERP-based brain-computer interface for communication; Authors: Marjolein van der Waal and Marianne Severens and Jeroen Geuze and Peter Desain; Year: 2012;

- [13] Lehman, M. M.; J. F. Ramil, P. D. Wernick, D. E. Perry, and W. M. Turski (1997). "*Metrics and laws of software evolution—the nineties view*". Proc. of 4th International Software Metrics Symposium (METRICS '97). pp. 20-32.
- [14] Eisenstadt, M., Domingue, J., Rajan, T., Motta, E.: Visual Knowledge Engineering. IEEE Transactions on Software Engineering, vol. 16, no. 10, pp. 1164-1177 (1990)
- [15] Zhang, K., Kong, J., Cao, J.: Visual Software Engineering. Wiley Encyclopaedia of Computer Science and Engineering (2008)
- [16] Lakin, F.: Visual grammars for visual languages. Proc. of the sixth National conference on Artificial intelligence AAAI'87, Vol. 2. AAAI Press, pp. 683-688 (1987)

8. Terminų ir santrumpų žodynas

NCI – nervų – kompiuterio sąsaja (angl. Neural computer interface).

BCI – smegenų – kompiuterio sąsaja (angl. Brain – computer interface).

HBCI – hibridinė smegenų-kompiuterio sąsaja (angl. Hybrid Brain-computer interface).

Elektroencefalograma – tai smegenų žievės elektrinio aktyvumas, kuris gali atvaizduoti įvairias sąmonės būsenas bei akių judesius ar raumenų įtempimą.

Elektrokardiograma – tai širdies veiklos diagrama.

Elektromiograma – tai raumenų elektrinio aktyvumo diagrama.

Elektroneurograma – tai elektros impulso sklaidimo greičio nervų skaidulomis diagrama.

ERP – su įvykiais susijęs potencialas (angl. Event-related potential).














Okulografija – akių judesių signalai.















NIA – (angl. Neural Impulse Actuator) – OCZ firmos gaminamas nervinius registruojantis prietaisas.




9. Priedai

Lentelė 5 Konceptai ir jų reikšmės

Konceptas	Reikšmė
	Konceptas apibendrinantis emocijų konceptus.
	Džiugesys. Linksmumas. Laimė.
	Nepasitenkinimas. Liūdesys.
	Nuostaba.
	Skausmas. Galvos skausmas.
	Šaltis.
	Šiluma. Karštis. Aukšta temperatūra.
	Konceptas apibendrinantis vietos konceptus.
	Namai. Pastatas. Gyvenamoji vieta.
	Ligoninė ar vieta, kurioje teikiama medicininė pagalba.
	Konceptas apibendrinantis veiksmus.
	Konceptas apibūdinantis bendravimą.
	Skambinti. Kviesti.

	Konceptas reiškiantis pagalbą.
	Skaityti. Skaitymas.
	Konceptas reiškiantis kasymo veiksmą.
	Konceptas apibendrinantis laiką.
	Šiandien.
	Rytoj. Ateitį.
	Vakar. Praeitį.
	Konceptas apibendrinantis daiktus.
	Maistas.
	Konceptas apibendrinantis su medicina susijusius daiktus.
	Greitoji medicinos pagalba.
	Medikamentai.
	Knyga.

	Stiklinė vandens. Gėrimas.
	Televizorius.
	Konceptas apibendrinantis žmogaus kūno dalis.
	Ranka. Rankos.
	Nugara.
	Galva.
	Širdis. Širdies plotas.
	Koja. Kojos.
	Plaučiai.
	Skrandis. Pilvas. Plotas aplink pilvą.
	Asmuo. Aš.
	Daktaras ar su medicina susijęs asmuo.
	Konceptas grupuojantis pagalbinius konceptus.
	Rodymas kairėn. Kreipimasis į kitą asmenį/save.

	Rodymas dešinėn. Kreipimasis į kitą asmenį/save.
	Draudimas, neigimas.
	Konceptas reiškiantis klausimą.
	Konceptas atitinkantis tarpą ar sakinio pabaigą.

9.1. Straipsniai

9.1.1. Evaluation of user fatigue in neural computer interface system

Evaluation of User Fatigue in Neural Computer Interface System

M. Vasiljevas¹, I. Martišius², T. Šumskas¹

¹Software Engineering Department,

²Computer Science Department,

Kaunas University of Technology,

Studentu 50, Kaunas Lithuania

Email: mindaugas.vasiljevas@ktu.lt

Abstract— The paper analyses analytical models of fatigue and their applicability to Physiological Computing Systems (PCS). For evaluation of fatigue we use a neural interface system (NIS) as an example of PCS for controlling an external physical entity (a robot) using EMG (electromyography) signals. We use the system to evaluate the state and performance of the users in performing control operation as well as to analyse dynamical changes of user's state (such as induced by fatigue) over time. The results suggest validity of classical model of muscle fatigue suggested by Banister et al.

Keywords— physiological computing, neural interface, EMG, fatigue.

1. INTRODUCTION

Physiological computing systems (PCS) use physiological data of its users as input for performing computational tasks [1]. Data is collected using physiological data sensors (e.g., electroencephalography (EEG), electrooculography (EOG), electromyography (EMG), etc.) attached to the user's body. Sensors directly measure signals triggered by the events occurring within the human body. Such input represents an additional channel of communication [2] between a user and a computer that allows a PCS to monitor, detect and respond appropriately to the physical, emotional and cognitive modalities of its users.

Since a PCS monitors the user's state, it creates a *bio-cybernetic feedback loop* [3] that allows to produce a more accurate representation of the user's context using information received from sensors attached to the human body. Such loop represents an innovative mode of human-computer interaction (HCI) where interaction is achieved by observing, analysing and responding to psychophysiological activity of a user [4]. The increase of the number of HCI modes beyond common audio, video, mechanical and tactile ones [1] promises extended capabilities of using computers for disabled users [5], as well as increasing engagement and immersion of common computer users (e.g., in computer games [6]), and increasing general intelligence of computer systems in smart environments such as Internet-of-Things.

However, the conscious use of PCS is hindered by many factors such as availability of low-cost high-quality sensors and

the need to develop more effective physiological signal processing and user state detection algorithms as well as steep learning curve for using PCS. One of the factors often neglected is fatigue (both mental and muscular one) which reduces accuracy and information transfer rate of the PCS communication channel and leads to input errors.

Fatigue is a gradually increasing subjective feeling of tiredness of a subject under load. Fatigue can have physical or mental causes. Physical fatigue is the inability of a muscle to maintain optimal physical performance [7]. Mental fatigue is a decrease in maximal cognitive performance resulting from prolonged periods of cognitive activity [8]. Mental fatigue leads to deterioration of cognitive functions as due to deterioration of cognitive functions user responses become slower and more error prone [9]. The result of fatigue is decreased productivity of using a PCS as well as critical errors in worst cases. Despite the importance of the fatigue factor there is little research of how to detect, measure and use it as factor for adapting the behaviour of a PCS.

In this paper we analyse analytical models of fatigue and their applicability to Physiological Computing Systems (PCS). For evaluation of fatigue we use a neural interface system (NIS) for robot driving control as an example of PCS that uses EMG as input. We use the system to evaluate the state and performance of the users in performing control operation as well as to research dynamical changes of the user's state (such as induced by fatigue) over time.

The structure of the remaining parts of the paper is as follows. In Section II we analyse the mathematical (analytical) models of fatigue. In Section III we describe the developed NIS system and the methodology of measurement. In Section IV we describe and discuss the experimental results. Finally in Section V we present conclusions and discuss future work.

2. OVERVIEW OF ANALYTICAL MUSCLE FATIGUE MODELS

Mathematical and analytical models provide a method for describing and predicting the effect of mental and muscular load on the performance characteristics of a human [10]. Analysing physical performance data one can identify and quantify different effects of loads such as increased performance (fitness or learning) and decreased performance (fatigue). Such models have been extensively researched in sports medicine [11], [12] as well as in applied physiology

[13], [14]. Consequently, there are two research streams: one is analysing the fitness and fatigue models affecting the sport performance of athletes [11-14], while another one aims to model the muscular response to stress.

The Banister model [11] and its various elaborations [12-14] is aimed to relate training loads to performance, taking into account the dynamic and temporal characteristics of load sequences over time. These effects can be described by two transfer functions: 1) a positive influence (i.e., muscular learning or fitness) that sums up all positive effects leading to an increase in performance, and 2) a negative function that summarizes effects leading to fatigue and having a negative influence on performance. The transfer function is as follows:

$$p_t = p_0 + k_a \sum_{s=0}^{t-1} e^{-(t-s)/\tau_a} w_s - k_f \sum_{s=0}^{t-1} e^{-(t-s)/\tau_f} w_s \quad (1)$$

where p_t is the modelled performance at time t ; p_0 is the initial performance level; k_a and k_f are the fitness and fatigue magnitude factor, respectively; τ_a and τ_f are the fitness and fatigue decay time constant, respectively; and w_s is the known training load per week (or day) from the first week of training to the week (or day) preceding the performance.

Calvert *et al.* [12] have proposed a model to quantify the training and performance relationship of a swimmer. The model was derived by fitting a mathematical equation and examining its parameters. A single training impulse elicited two fitness responses that increase performance, and a fatigue response that decreases performance. The final form of the Calvert's model was presented as follows:

$$p(t) = \left[e^{-\frac{t}{\tau_1}} - e^{-\frac{t}{\tau_2}} - Ke^{-\frac{t}{\tau_3}} \right] w(t) \quad (2)$$

where $w(t)$ is the training impulse, $p(t)$ is performance, τ_1 and τ_2 are the time constants associated with the two fitness functions and τ_3 is the time constant associated with fatigue, and K is the fatigue coefficient specific to the individual, and t is the day of the training impulse.

Morton *et al.* [13] simplified the Calvert's model (Eq. 2) to two components (one for fitness and one for fatigue) because the second fitness component was not statistically supported. The Morton's model is as follows:

$$p(t) = \left[e^{-\frac{t}{\tau_1}} - Ke^{-\frac{t}{\tau_3}} \right] w(t) \quad (3)$$

Busso *et al.* [14] defined how fitness and fatigue are affected by a training input as follows:

$$\begin{aligned} g(t) &= g(t-1)e^{\frac{-1}{\tau_1}} + w(t), \\ h(t) &= h(t-1)e^{\frac{-1}{\tau_2}} + w(t), \end{aligned} \quad (4)$$

where $g(t)$ and $h(t)$ are arbitrary fitness and fatigue response levels at the end of day t , and τ_1 and τ_2 are decay constants of each input.

Eq. (4) were then combined to form a simple linear difference equation

$$p(t) = k_1 g(t) - k_2 h(t), \quad (5)$$

where k_1 and k_2 are weighting factors for fitness and fatigue, respectively.

Summarizing, the fatigue models discussed here are based on the exponential decay function that is widely used to describe natural phenomena such as heat transfer between the object and its medium, rate of enzyme-catalysed chemical reactions, fluid dynamics, metabolization of drugs in patients. These examples provide a logical foundation for application of exponential decay function for fatigue modelling. However the fatigue models also have been criticized for imprecision and low accuracy due to variability of their parameters [15]. Furthermore, the models also need verification in the context of physiological computing, where signals of human body are usually registered under normal conditional rather than considerable strain. Also the problem of mental fatigue is usually ignored, though over time it leads to decrease of performance in PCS.

3. EXPERIMENTAL NEURAL INTERFACE SYSTEM

We use the NIS (first described in [16]) that has these components (see Fig. 1):

- Data Acquisition Module – reads the neural data from the sensors via USB;
- Data Processing Module – processes data from Data Acquisition Module for noise removal and feature identification;
- Training Module – produces a classification model for identification of mental states;
- Classification Module – identifies mental states in the data using a classification model; and
- Device Control Module – maps mental states to control commands and sends them to an external device.

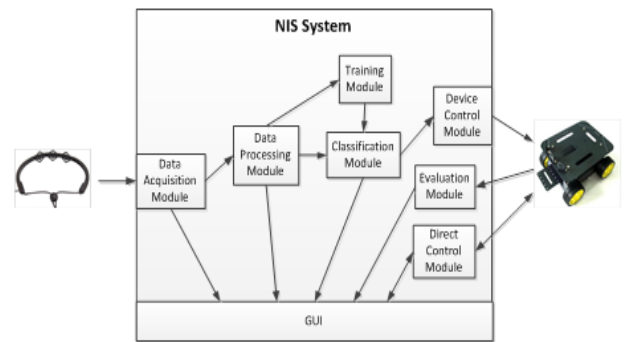


Fig. 1. Components of the Neural Interface System (NIS) [16]

The Data Acquisition Module is implemented to receive data from the OCZ Neural Impulse Actuator (NIA) (see Fig. 2). The NIA is a NIS device that has 3 sensors and uses a USB connector to connect to a computer. The NIA captures three types of signal from the brain and forehead: EEG data (the neuronal discharges in the brain), EOG data (positional differential between the front of retina and the retinal pigmented epithelium which changes relative to the eye orientation), and EMG data (neuro-muscular signals along with electrical discharges resulting from depolarization of the

muscle cells). The data is wrapped in packets and delivered via USB. Raw data is provided unprocessed, so we use our own methods to map mental states to control commands.



Fig. 2. Neural signal registration using OCZ NIA device

The aim of the training module is to capture the individual characteristics of the neural signals of the subject, who uses the data. The component manages the training session, which consists of flashing symbols shown on the screen while the neural data received from the data acquisition module is analysed and the training dataset is constructed. Each symbol represents a specific control command of an external robotic device such as “Move forward” or “Move right”. The control command is then encoded as a class of data, while the data received during visual presentation of the command (i.e., neural feedback) is saved as the features of the class.

For externally controlled physical entity we use the Arduino 4WD Mobile Platform provides a 4 wheel drive system complete with ATmega328 microcontroller board and 4 DC Motors. The platform can be connected to a computer with a USB cable or powered using the AC-to-DC adapter or battery. It has four degrees of freedom (forward, backward, left, right).

4. EXPERIMENT

The experiments we performed with 5 healthy subjects aged between 20 and 30 years. For each person a list of 120 task was generated randomly. The task list included 40 tasks to issue the commands „left“, „right“ and „forward“, respectively. The subjects were instructed to issue commands as follows: “left“ – to blink with left eye, „right“ – to blink with right eye, „forward“ – to stress the forehead muscles. The command „stop“ was not generated and registered during the experiment because it is executed when a subject is in complete standstill (without straining facial muscles). This condition is natural for most persons and requires no effort, i.e., it does not lead to muscular fatigue. All commands were distributed randomly on a task list. For each task execution 5 seconds were allocated. As subject initiates the given command, another person (the observer) observed whether the behaviour of the robot corresponds to the command issued by the subject and the results registered. Each experiment was repeated 5 times with a 30 min break for resting. The results of the experiment were used to calculate the accuracy of robot control using the following formula:

$$A = \frac{n - f}{n} \quad , \quad (6)$$

here A – is the accuracy of control, n – is the number of tasks, and f – is the number of incorrectly executed commands.

The results of the experiments are presented in graphically in Fig 3.

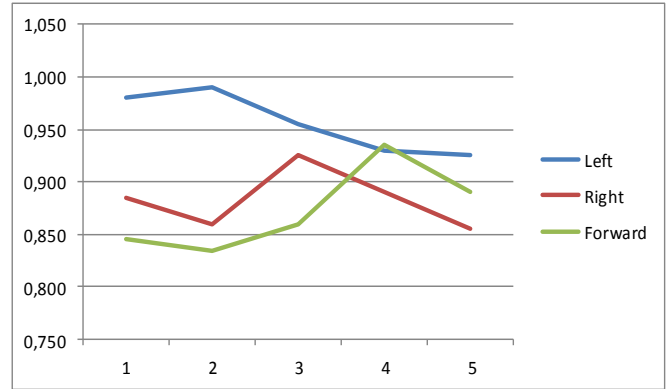


Fig. 3. Accuracy results for robot control commands

To model user fatigue during the robot control experiment, we assume that control accuracy follows the classical exponential decay model with two components: the positive one is corresponding to learning (training), and the negative one is corresponding to user fatigue (as defined in the Banister *et al.* [11] model). We can see from the Fig. 3 that the trend of classification accuracy corresponds to our fatigue model. Accuracy first increases due to user learning processes, but finally starts to decrease due to user fatigue.

The proposed model used to fit experimental data is as follows:

$$A(t) = A_0 + \lambda \cdot e^{-t/\tau_1} - \varphi \cdot e^{-t/\tau_2} \quad , \quad (7)$$

here $A(t)$ is accuracy, A_0 is base accuracy, λ is learning parameter, φ is fatigue parameter, t is time, τ_1 and τ_2 are decay parameters for learning and fatigue respectively.

The results of fitting are presented graphically with 0.95 confidence intervals in Fig. 4. Note that base accuracy is modelled as random accuracy ($A = 0.5$) at $t = -1$ as well as at a final time period $t = -10$.

We assume that at first, the user can not issue correct control commands because he/she has not been trained yet. Finally, the user is also unable to issue correct control commands due to extreme fatigue.

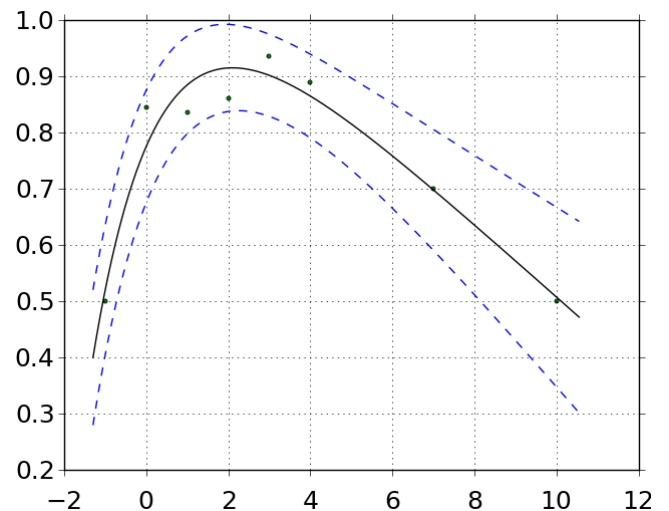


Fig. 4. Results of data fitting (accuracy vs. time periods)

5. CONCLUSION

Our initial experiments with evaluating user fatigue when a subject is working with the experimental Neural Computer Interface (NCI) system suggest that the classical model of Banister *et al.*, first suggested to evaluate performance of sport's athletes, is applicable to the domain of physiological computing as well. We have analysed the accuracy of tasks performed in controlling an external robotic device using the EMG-based NCI system. While initial effort leads to improvement of task performance (the learning effect), more prolonged user effort leads to fatigue, both mental and muscular, because forehead muscles are used for control. Experimental data was fitted to the proposed user fatigue model adapted from Banister *et al.* The proposed model has been shown to be valid with 0.95 confidence.

Further research is needed in analysing long-term effects of fatigue which may not be covered by the presented experiment.

6. ACKNOWLEDGEMENT

The authors would like to acknowledge the contribution of the COST Action IC1303 - AAPELE, Architectures, Algorithms and Platforms for Enhanced Living Environments.

REFERENCES

- A. Schroeder, and M. Wirsing, "Developing Physiological Computing Systems: Challenges and Solutions", *Software Engineering 2012*, pp. 21-36, 2012.
- L. J. Hettinger, P. Branco, L. M. Encarnaco, and P. Bonato, "Neuroadaptive technologies: Applying neuroergonomics to the design of advanced interfaces", *Theoretical Issues in Ergonomic Science*, vol. 4, no. 1-2, pp. 220-237, 2003.
- N. B. Serbedzija, S. H. Fairclough, "Biocybernetic loop: From awareness to evolution", *IEEE Congress on Evolutionary Computation, IEEE*, pp. 2063-2069, 2009.
- J. Allanson, and S. H. Fairclough, "A research agenda for physiological computing", *Interacting with Computers (IWC)*, vol. 16, no. 5, pp.857-878, 2004.
- H. Huang, Y. Chen, W. Xu, H. Sundaram, L. Olson, T. Ingalls, T. Rikakis, and J. He, "Novel design of interactive multimodal biofeedback system for neurorehabilitation", in *Conf. Proc. IEEE Eng .Med. Biol. Soc.*,1:4925-8, 2006.
- L. E. Nacke, M. Kalyn, C. Lough, and R. L. Mandryk, "Biofeedback game design: using direct and indirect physiological control to enhance game interaction", *Proc. of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*, ACM, New York, NY, USA, pp. 103-112, 2011
- J.A. Hawley, and T. Reilly, "Fatigue revisited", *Journal of sports sciences*, vol. 15, no. 3, pp. 245-6, 1997.
- S. Marcora, "Mental fatigue impairs physical performance in humans", *Journal of Applied Physiology*, vol. 106, no. 3, pp. 857-864, Jan. 2009.
- M. K. Scheffers, D. G. Humphrey, R. R. Stanny, A. F. Kramer, and M. G. Coles, "Error-related processing during a period of extended wakefulness", *Psychophysiology*, vol. 36, pp. 149-57, 1999.
- T. Taha, and S. G. Thomas, "Systems modelling of the relationship between training and performance", *Sports Med.*, vol. 33, no. 14, pp. 1061-73, 2003.
- E. W. Banister, T. W. Calvert, M. V. Savage, and A. Bach, "A system model of training for athletic performance", *Australian Journal of Sports Medicine*, vol. 7, pp. 170-176, 1975.
- T. W. Calvert, E. W. Banister, M. V. Savage, et al., "A systems model of the effects of training on physical performance", *IEEE Trans Syst Man Cybern*, vol. 6, no. 2, pp. 94-102, 1976.
- R. H. Morton, J. R. Fitz-Clarke, and E. W. Banister, "Modeling human performance in running", *J Appl Physiol*, vol. 69, no. 3, pp. 1171-7, 1990.
- T. Busso, H. Benoit, R. Bonnefoy, L. Feasson, and J. R. Lacour, "Effects of training frequency on the dynamics of performance response to a single training bout", *Journal of Applied Physiology*, vol. 92, pp. 572-580, 2002.
- P. Hellard, M. Avalos, L. Lacoste, F. Barale, J. C. Chatard, and G. P. Millet, "Assessing the limitations of the Banister model in monitoring training", *J Sports Sci.*, vol. 24, no. 5, pp. 509-20, May 2006.
- I. Martisius, M. Vasiljevas, K. Sidlauskas, R. Turcinas, I. Plauska, and R. Damasevicius, "Design of a Neural Interface Based System for Control of Robotic Devices", *Proc. of the 18th Int. conf. on Information and Software Technologies, ICIST 2012, Kaunas, Lithuania, September 13-14, 2012*. Springer-Verlag, CCIS 319, pp. 297-311, 2012.

9.1.2. Development of Concept-Based EMG-Based Speller

Development of Concept-Based EMG-Based Speller

Desarrollo de un deletreador basado en Conceptos basados en EMG

Robertas Damaševičius, Mindaugas Vasiljevas & Tomas Šumskas

*Software Engineering Department, Kaunas University of Technology, Studentu str. 50, Kaunas, Lithuania, +370 37 300 353
robertas.damasevicius@ktu.lt, mindaugas.vasiljevas@ktu.lt, tomas.sumskas@ktu.lt*

Received: Received in revised form: Accepted:

Abstract

Physiological computing is a paradigm of computing that uses physiological data of users as input during computing tasks in an Ambient Assisted Living (AAL) environment. By monitoring, analyzing and responding to such inputs, Physiological Computing Systems (PCS) are able to respond to the cognitive, emotional and physical states of users. A specific case of PCS is Neural Computer Interface (NCI), which uses electrical signals governing muscular activity (EMG data) of a user to establish a direct communication pathway between the user and a computer. We present taxonomy of speller application parameters, propose a model of PCS, and describe the development of the EMG-based speller as a benchmark application. We analyze and develop an EMG-based speller application with a traditional letter-based as well as visual concept-based interface. Finally, we evaluate the performance and usability of the developed speller using empirical (accuracy, information transfer speed, input speed) metrics.

Keywords: Physiologic computing, NCI, EMG, speller, user interface.

Resumen

La computación fisiológica es un paradigma de la computación que usa los datos de los usuarios como entradas durante las tareas computacionales en un Ambiente de vida cotidiana soportado por computadores (AAL). Monitoreando, analizando y respondiendo a dichas entradas, los Sistemas de Computación Fisiológica pueden responder al estado cognitivo, emocional y físico de los usuarios. Un caso particular es el de la interface de Computación Neuronal (NCI), que usa señales eléctricas para manejar la actividad muscular del usuario estableciendo una comunicación directa entre el usuario y el computador. Se presenta una taxonomía de parámetros de aplicación de deletreo, proponiendo un modelo de PCS y describiendo el desarrollo de un deletreador basado en EMG. Se analiza y desarrolla una aplicación con un sistema basado en letras tradicionales y una interfaz visual. Finalmente, se evalúa el desempeño y usabilidad del sistema desarrollado.

Keywords: Computación fisiológica, NCI, EMG, deletreador, interfaz de usuario.

1 Introduction

Today's computer systems are failing to satisfy the increasing expectations of everyday users. While the output of computer systems features a multimedia communication channels, the input to computers is still fairly limited to mechanical (keyboard, mouse), audio (speech) and tactile inputs [1]. Such limitations raise barriers for people with major or minor disabilities such as elderly people with motor impairments. Considering predictions of the demographic changes in society and the need to improve the quality of daily life for humans in different periods of their life, new concepts and methods of human-computer interaction (such as Ambient Assisted Living (AAL) [2] systems) must be researched that efficiently address the accessibility problems in human interaction with software applications and services while meeting the individual requirements of the users in general, including disabled and elderly people.

Physiological computing is a paradigm of computing that uses physiological data of users as input during computing tasks [3]. By monitoring, analysing and responding to such inputs, physiological computing systems (PCS) are able to monitor and respond to the cognitive, emotional and physical states of users in real time. Examples of such physiological data are Electroencephalography (EEG), Electrooculography (EOG), Electromyography (EMG), etc. The state of a user is captured using sensors attached to the body, and could be used for offering assistance if the user is frustrated or unable to perform the task due to excessive mental workload, adapting the level of challenge to sustain or increase engagement, or incorporating an emotional display element into the user interface [4].

Specific cases of PCS are neural computer interface (NCI) and brain computer interface (BCI). BCI uses brain activity (EEG) of a user to establish a direct pathway between the brain and computer. The concept is particularly suited to the needs of the handicapped as well for smart environments. NCI is similar to BCI in methods used as well as in applications, however it uses the surface EMG signals to establish an interface between human peripheral neural system and computers by recording electrical signals governing muscular movements of a subject.

A speller is a typical example of PCS, which is a benchmark application for BCI and NCI methods [5]. The speller aims to help persons unable to activate muscles traditionally used in communication (hands, tongue) to spell words by utilizing their neural activity. Usually, spellers use signal amplitude information, however, integrating it with signal preprocessing methods such as noise reduction methods (e.g., shrinking functions [6], nonlinear filter operators [7]), and user intent prediction techniques can improve the results [8]. Due to the noise present in the physiological signals, the detection of control signals requires using efficient digital signal processing techniques [9, 10, 11]. Furthermore, the requirements for real-time systems prohibit the use of highly complex computations and demand the result to be delivered just in time to be used by the user [12].

In this paper, we continue our previous work [13, 14] and describe a model of PCS for EMG-based applications, analyze the requirements for developing interfaces for impaired users and visual interfaces of known speller applications, and describe the development of an EMG speller as a typical benchmark application.

2 Analysis of requirements for speller applications

The requirements for speller application can be categorized at different levels depending upon the physical abilities of its users [15]: 1) Users with no physical disability, who may use NCI for entertainment or other conditions where physical movement is restricted. 2) Users with minor impairments (such as older persons). 3) Users with severe physical disabilities, who may wish to use NCI as a secondary input. 4) Users who have limited muscle control and may need to use NCI as a method for communication. First, the speller must follow general requirements for smart systems to be integrated into the AAL environments. Next, the specific requirements for impaired users must be followed. Impaired users need assistance such as automatic learning of user's behavior to estimate his/her current needs. Humans often make mistakes in interacting with machines, so for any human-operated system user interfaces should be designed such that prevent errors whenever possible; deactivate invalid commands; make errors easy to detect and show users what they have done; and allow undoes, reverse, correct errors easily [16]. For smart systems, the following principles (also called "operational modes") of Humanistic Intelligence Framework [17] must be satisfied:

1) **Constancy:** the interface should operate continuously to read signals from human to computer and to provide a constant user-interface.

2) **Augmentation:** the primary task is increasing the intelligence of the system rather than computing tasks.

3) **Mediation:** the interface mediates between human senses, emotions and perceptions and acts as information filter by blocking or attenuating undesired input to decrease negative effects of interaction (such as fatigue, information overload, etc.) and to increase positive effects (such as user satisfaction) by amplifying or enhancing desired inputs.

According to Lopes [18], user interface for persons with disabilities must: support user variability allowing to provide the means to adapt to user-specific requirements; support of a wide range of input devices and output modes; provide minimal user interface design; promote interaction and retain user attention on the tasks; and provide strong feedback mechanisms that may provide rewarding schemes for correct behavior (results). The requirements for interfaces for impaired users can be formulated as follows [2]: 1) **Limited access to details:** complex and vital details of the system have to be hidden to avoid user overwhelming and trapping. 2) **Self-learning:** detected common patterns in the behavior of the user should be used to automatically create rules or shortcuts that speed and ease up the use of the system. 3) **System interruption:** Impaired users have in most cases no idea how the system is working, therefore easy cancellation of system's activities must be ensured.

3 Overview of speller systems and interfaces

The research in developing and improving speller systems focuses on improving accuracy of spelling, increasing speed of information transfer, developing usable and effective speller interfaces, and combining EEG/EMG based input with input automation techniques such as word complete and automatic correction of misspellings.

The speller implementations can be characterized by:

Type of data: EEG [19], EMG [20], ECoG [8], EOG [21].

Type of analyzed signal: P300 event-related potentials (ERPs), which are series of peaks and troughs appearing in the EEG in response to occurrence of a discrete event, such as presentation of a stimulus or psychological reaction to a stimulus [8], Error-related Potentials (ErrPs) generated by the subject's perception of an error [22], Steady-state visual evoked potential (SSVEP), which are signals that are natural responses to visual stimulation at the same (or multiples of) frequency of the visual stimulus [23].

Modality: *Auditory:* the rows and columns of the letter matrix are represented by different sounds, such as spoken numbers [24] or environmental sounds. *Visual:* subjects direct their eye gaze toward the letter they want to select. There are two cases: overt attention when eye gaze is directed toward the target letter, and covert attention when eye gaze is directed at a central fixation point [25].

Interface:

Single character (or Linear) speller: all letters are shown and each letter is flashed individually until letter selection is done [26].

Matrix Speller: All letters are arranged in a matrix. First, speller flashes an entire column or row of characters. Then, single letters are flashed in a sequence, and can be selected [27]. Different matrix sizes can be used, e.g., a 6x6 matrix containing all 26 letters of the alphabet and 10 digits (0-9), or even a full QWERTY keyboard [23].

Lateral single-character is a single-character paradigm comprising all letters of the alphabet following an event strategy that significantly reduces the time for symbol selection [28].

Checkerboard Speller [29]: the 8x9 matrix is virtually superimposed on a checkerboard, which the participants never actually see. The items in white cells of the 8 x 9 matrix are segregated into a white 6 x 6 matrix and the items in the black cells are segregated into a black 6 x 6 matrix. The items in the first matrix) randomly populate the white or black matrices, and the users see random groups of six items flashing (as opposed to rows and columns in Matrix Speller). Such layout controls for adjacency-distraction errors, because the adjacent items cannot be included in the same flash group.

Hex-o-Spell: speller consists of six circles that all have the same distance to the point of fixation. The circles are flashed while users direct their attention to one of the circles. First, the circle with the desired group of letters is selected. Second, letters are redistributed over the circles and the target letter is selected [30].

Frequency-based layout accounts for the relative frequency of character occurrence in a language [31]. It has a virtual keyboard with 32 symbols surrounded by five boxes flickering at different frequencies. These boxes correspond to commands for navigating the cursor, and for selecting the intended character. The application starts with a cursor in the central position corresponding to the most frequent character in English (i.e., "E"). Letters with the higher frequency of occurrence are positioned closer to the center while the less frequent ones are further away. The user can navigate the cursor to the desired letter and confirm his/her choice with the "Select" command. The further the character is located from the center, the more command selections (cursor movements) are required.

Stimulus type: the way each individual character changes (e.g., flashing, color change, etc.). Rapid serial visual presentation (RSVP) is a method of displaying information

(generally text or images) in which the text is displayed word-by-word in a fixed focal position [32].

Stimulus rate: the speed at which individual characters change.

Stimulus pattern: grouping of symbols in interface (e.g., QUERTY or DVORAK layouts in a virtual keyboard).

Character set (alphabet): includes all letters of the alphabet as well as some additional symbols (numbers, separation marks, etc.).

Intelligence techniques: additional techniques for improving accuracy of the system and rate of communication such as using language model [33], word autocomplete, spelling correction or word prediction.

The result of the analysis can be considered as taxonomy of speller application parameters, which can be used for developing new speller applications. Next we discuss the model of a PCS and its application to developing the EMG-based systems.

4 Model of a physiological computing system

The model of physiological computing system (PCS) proposed in this paper is based on the "operational modes" of the Humanistic Intelligence Framework [17]:

- 1) **Constancy:** PCS should operate continuously to read signals from human and provide a constant user interface.
- 2) **Augmentation:** the primary task is increasing the intelligence of the system rather than computing tasks.
- 3) **Mediation:** PCS mediates between human senses, emotions and perceptions and acts as information filter by blocking or attenuating undesired inputs to decrease negative effects of interaction (fatigue, information overload) as well as to increase positive effects (e.g., user satisfaction) by amplifying or enhancing desired inputs.

The proposed model is summarized in Figure 1 and has three levels of information processing.

- 1) On the lowest level, the physiological signal is sampled into a data stream of physiological data. Downsampling can be used to decrease amount of data and increase information processing speed at higher levels.
- 2) On the intermediate level, data is aggregated and events corresponding to specific patterns of data are generated. Machine learning techniques such as artificial neural networks may be used to recognize such events and generate decisions.
- 3) On the highest level, decisions are processed and used to generate control commands for external systems.

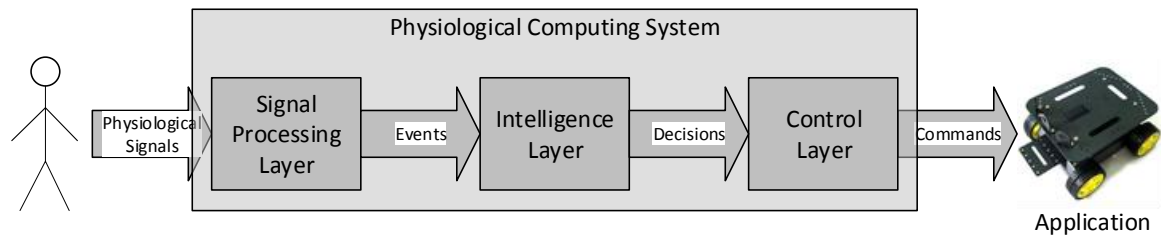


Figure 1. Model of physiological computing system. Source: The authors.

5 Development of EMG speller

5.1 Components and architecture

A NCI system generally comprises the following components: (1) a device that records the muscular activity signals; (2) a signal preprocessor that reduces noise and artifacts; (3) a decoder that classifies the de-noised signal into a control commands for (4) an external device or application (e.g., a robotic actuator, a computer program etc.), which provides feedback to the user [34].

Our speller application has three layers as follows: 1) on the lowest layer, the physiological signal is sampled into a data stream of physiological data. Downsampling can be used to decrease amount of data and increase information processing speed at higher levels. 2) On the intermediate layer, data is aggregated and events corresponding to specific patterns of data are generated. Machine learning techniques such as artificial neural networks may be used to recognize such events and generate decisions. 3) On the highest layer, decisions are processed and used to generate control commands for external applications (systems).

Architecture of the speller consists of 6 main components: *MainReader* – system module responsible for control of data reader which is selected to use. *ReaderAPI* – public external interface module. All third-party modules must implement this component for full system integration. *MainController*– system module responsible for selected control module (executes commands). *NiaReader*– third-party module implemented for the “OCZ NIA” data reader device. *SpellingSquare*– third-party module implemented for text input in the symbol matrix using EMG-based commands. The dashed rectangle separates system components (inside the dashed rectangle) from external components, which are either the sensor controllers (EMG readers) or actuator controllers (software or hardware applications such as robots). The speller system is based on *Java NetBeans* framework. The speller was developed with considering its future extension and maintenance so that external components are easy to add or remove.

5.2 Control

There are two types of control commands of the speller: “Select” command – selects a column or types a symbol of that column. “Cancel” command – exits the selected column or deletes the selected symbol. Those control commands can be initiated by the movements of facial muscles. In practice, eye blinks are used to generate each control command (left eye blink for “select” and right eye blink for “cancel”). The user can see the EMG signal feedback in EMG signal view area (see Figure 2). The particular control command is performed when the amplitude of the EMG signal is higher than the specified threshold value. The thresholds are marked as yellow horizontal lines in the EMG signal view area. The upper threshold indicates the “select” command, and the lower threshold indicates the “cancel” command. Threshold values can be adjusted using threshold setting sliders. The signal view of EMG, while spelling the word “hello”, is presented in Figure 2. In Figure 2 (left), the word “hello” is spelled without mistakes. In Figure (right), the spelling contains has a few mistakes. For correction of those mistakes cancellation commands must be performed. The spikes indicate the “select” command. One trial (selection of one character) contains two

positive signal spikes, the first spike is for column selection, the second for letter selection in the corresponding column.

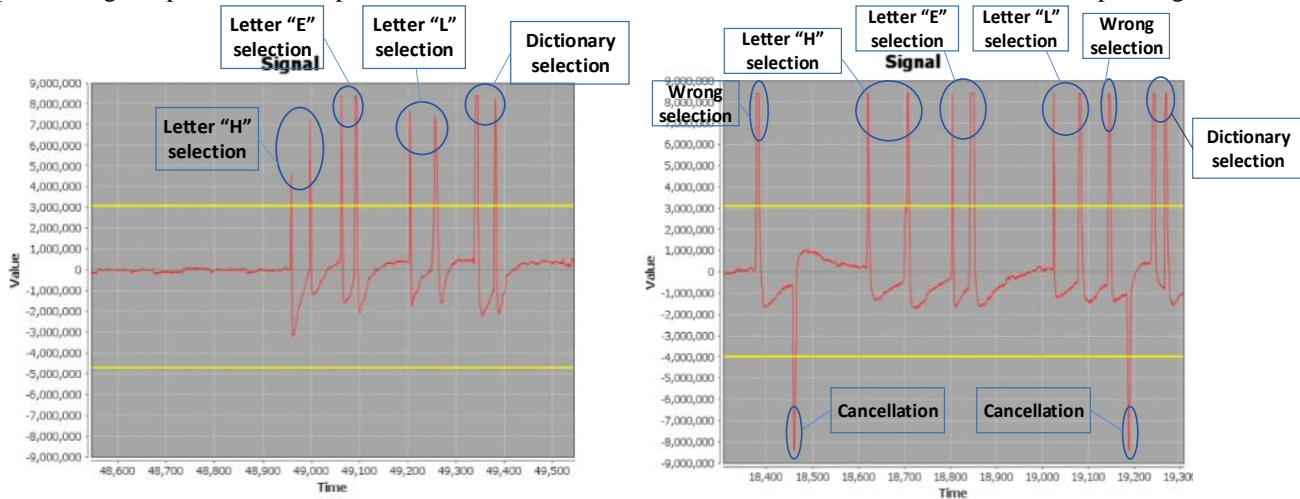


Figure 2. Signal view of spelling word “hello”. Left: no spelling mistakes were made while three characters (“hel”) selected from symbol matrix. Right: two spelling mistakes were made, and after each wrong selection cancellation command was performed. In both cases, dictionary selection was made to complete the word. Source: The authors.

5.3 Traditional speller interface

The developed EMG speller has two different user interfaces. One is traditional matrix-based speller interface presented in Figure 3a. Another one is a novel concept-based interface presented in Figure 3b. The most important part of the visual interface is symbol matrix. The matrix is adaptable so that various symbols (including special or national) could be added into the matrix. The red-colored column indicates the current position of the speller cursor. The cursor moves coherently from column to column until user activates the “select” command. Next, the cursor moves through each symbol in the particular column. After another “select” command, the particular symbol is selected. That symbol appears in the output area (see Figure 3). The speller cursor moves by the step which varies from 500 to 1500 milliseconds. The step value depends on the number of mistakes the user does. A smaller number of mistakes mean the faster speed of cursor movement. The mistake is considered as the “cancel” command.

The first row of the speller symbol matrix contains a dictionary selection. This selection allows to enter the dictionary. When few symbols or a stem of the word is written dictionary gives the opportunity to complete the particular word faster. The system logs dictionary selections thus common used words are at the top of the dictionary, thus the dictionary adapts to the user.

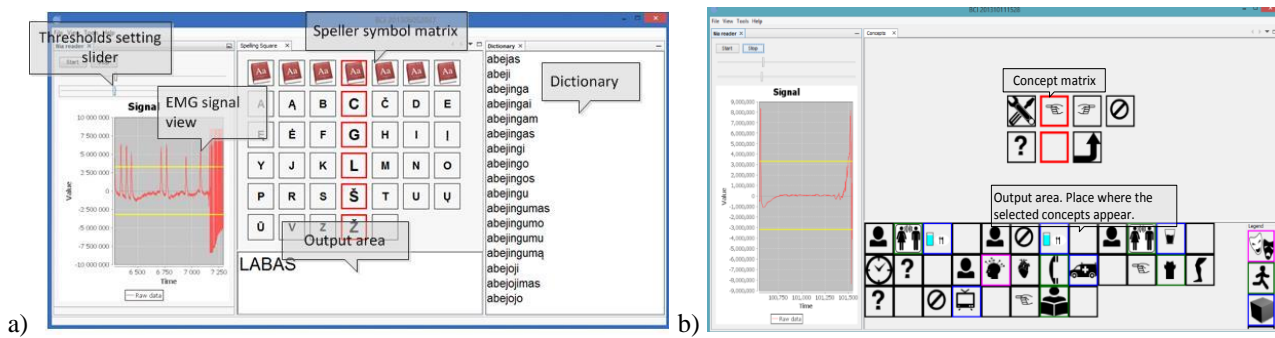


Figure 3. Interfaces of EMG speller: a) traditional, and b) visual concept based. Source: The authors.

5.4 Visual concept-based speller interface

We also have implemented a completely different interface of the EMG speller. Traditional spellers use common letters of an alphabet rearranged in different layouts. We have implemented a visual concept based interface that is based on graphical symbols (graphemes) of a visual communication language. Visual language is a form of communication that uses visual elements as opposed to a formal written (textual) language to convey meaning or an idea. Visual language uses pictograms or ideograms to signify the concepts that are communicated. Pictograms are pictures that resemble what they signify, and represent a concept, object, activity, place or event by graphical illustration. An ideogram is a graphical symbol that represents an idea, rather than a group of letters arranged according to the phonemes and grammar of a spoken language, as is done in textual languages.

Visual symbols form part of our daily lives through their use in medication, transport, computers, etc., because they indicate in a concise and easily understandable form places, directions, actions or constraints on actions in either the real world or virtual space. Thus, visual symbols can be used in a number of situations in which textual messages are not possible or adequate due to context or user based constraints. The development of software for people with impaired cognitive or motor capabilities requires by the developers to take into account the high-level cognitive processes and mental models of a solution of a domain-specific

problem (such as expressing basing needs of an impaired user), which may contribute to the ease of use of the developed application by its potential users.

While textual (letter-based) languages are good for expressing all kinds of human communication, they require many typing effort for human-computer communication, which for specific groups of users such as users with impaired motor capabilities may be a tiresome burden. A visual concept-based language allows to express high-level concepts succinctly using a notation tailored to a set of specific user problems. Such languages could be tailored towards a specific domain, and could be based only on the relevant concepts and features of that domain. Therefore, visual languages can be considered as a medium of communication that allows to bridge the gap between the mental models of users and the domain systems and, consequently, to cut the distance (both in terms of effort required and speed of communication) between communicating parties in AAL applications.

A snapshot of the developed visual concept-based interface of the EMG speller is given in Figure 3b. The interface is organized using a hierarchical structure. It consists of symbol matrixes connected with each other by references. Each reference is represented as an icon of particular domain. Currently, we have included visual symbols from 8 main concept domains: emotion domain, location domain, action domain, time domain, object domain, body part domain, person domain and special symbol domain. Most of visual icons we use are adopted from The Noun Project (<http://thenounproject.com/>), while the remaining ones are custom-built. Each domain matrix as well as the root matrix can be extended easily by adding new icons (concepts) to the particular domain matrixes. Also each icon (concept) could contain a reference to the specific subdomain matrix. The taxonomical tree of concept matrixes is summarized in Figure 4.

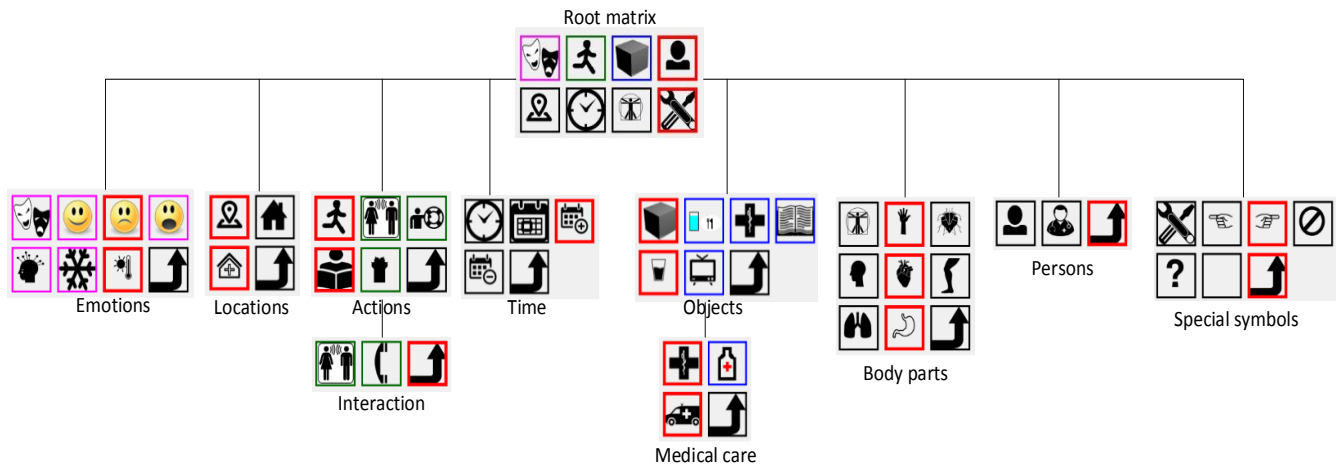

























Figure 4. The tree of concept matrixes. Source: The authors.

The meaning of visual concepts is presented in Table 1.

Table 1.

Meaning of visual symbols. Source: The authors.

	Return sign brings back cursor to the root matrix.
	Emotions/feelings:
	Happy
	Sad
	Amazed
	Pain
	Cold
	Hot/warm
	Location (current location):
	Hospital
	Home/house
	Actions:
	To read a book
	To interact:
	To call
	To scratch
	To help
	Time:
	Past

		Today
		Future
	Objects:	
		Drink
		Meal
		TV
	Medical care:	
		Ambulance
		Medicine
		Book
	Body parts:	
	Head	
	Hand	
	Heart	
	Stomach	
	Back	
	Leg	
	Person (Me):	
	Medic/doctor	
	Special symbols:	
		Question sign
		Left pointer to object or subject
		Right pointer to object or subject
		“Not” sign

6 Experimental results

6.1 Experiment with text-based interface

The experiments we performed with 5 subjects (3 males), aged 24–54 (mean = 33) year. Subjects did not have any neurological abnormalities, reported normal or corrected to normal vision, and did not use medication. All subjects gave informed consent prior to the experiment.

The EMG data was recorded using OCZ Neuro Impulse Actuator equipment. Visual stimuli were presented on a 13.3" size TFT LCD screen with 1360 × 768 pixel resolution and a refresh rate of 60 Hz. Subjects were seated in front of a table. The screen was in the middle of the table at a distance of approximately 100 cm from the subject. The size of each character was 1.5 × 1.5 cm (0.86 × 0.86° visual angle) and the entire speller matrix was 9.5 × 13 cm (5.44 × 7.42° visual angle).

Stimuli consisted of intensifications of the rows and columns in sequential order. Intensification was achieved by increasing the size of all characters in the row or column with a factor 500 for 1500 ms.

A trial is defined here as spelling of one character. All trials started with the speller being displayed on the screen, together with an instruction indicating which letter to select. Each stimulation sequence was followed by feedback on the screen, showing which letter or group of letters had been selected.

Three text paragraphs were given to the experiment participants. Their task was to input the proposed text paragraphs using speller. All text paragraphs were presented in Lithuanian language. The first text paragraph contained 126 characters and its content covered a daily conversation. The second text paragraph contained 111 characters and its content covered a scientific speech. The third text paragraph contained 120 characters and covered a scientific speech with mathematical equations. Each experiment participant repeated the experiment 4 times. The average accuracy, input speed and bit rate values were calculated.

Quantitatively, the performance of speller application can be evaluated using accuracy, information transfer speed and input speed metrics. Accuracy is calculated as the percentage of correct decisions during the symbol selection process. Bit rate (or information transfer rate) indicates how much information can be communicated per time unit (calculated using the Wolpaw's formula [35]). Finally, input speed is measured as the average time required to enter a set of benchmark texts.

The experimental results are presented in Table 2 and Figures 5, 6 & 7. Our results are comparable with the results achieved by other authors.

Accuracy values of the BCI/NCI spellers achieved by other authors (Table 2) are within 80-95% range (82.77% using ECoG [8], 87.58% using SSVEP-based BCI [24], 87.8% for EOG-based speller [21], 91.80% [36], 94.8% for RVSP based speller [32]).

The information transfer rate (aka bit rate) of the BCI/NCI-based speller applications achieved by other authors are within 7-41 bits/min (7.43 bits/min [37], 17.13 bits/min [38], 19.18 bits/min [36], 11.58-37.57 bits/min [39], 40.72 using SSVEP based BCI [23], 41.02 using ECoG [8]).

The symbol input speed of the BCI/NCI-based speller applications achieved by other authors are within 1-12 CPM (1.43 CPM for RSVP based speller [32], 4.33 CPM [36], 4.91 CPM [39], 9.39 CPM using SSVEP based BCI [23], 12.75 CPM [40]).

Table 2.

Evaluation of speller application. Source: The authors.

Quantitative metric	Average Value	Peak value
BASIC SETTINGS		
Accuracy	96.29	98.25
Information transfer rate	34.78	41.83
Input speed	6.37	7.57
ADAPTABLE STIMULUS RATE		
Accuracy	88.61	93.64
Information transfer rate	42.53	49.79
Input speed	8.19	9.60
WITH DICTIONARY		
Accuracy	92.65	96.06
Information transfer rate	43.55	49.26
Input speed	8.22	9.35
WITH ADAPTABLE STIMULUS RATE AND DICTIONARY		
Accuracy	89.16	92.53
Information transfer rate	58.69	65.53
Input speed	11.35	12.42

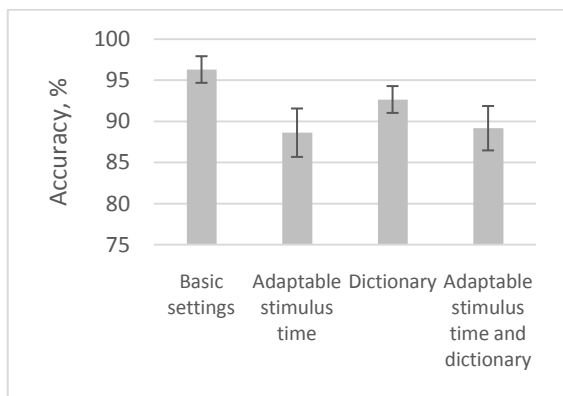


Figure 5. Accuracy of character input. Source: The authors.

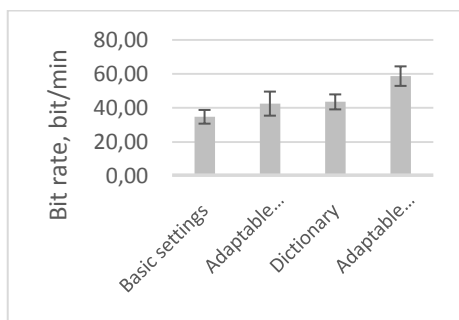


Figure 6. Information transfer rate. Source: The authors.

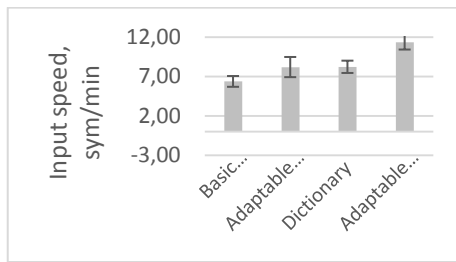
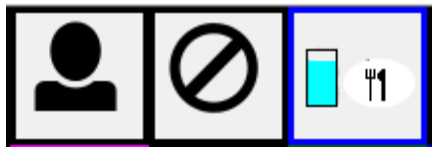


Figure 7. Input speed. Source: The authors.

6.2 Experiment with visual concept based interface

The experiment with the visual concept-based interface was performed under the same conditions as the experiment with text-based interface. The participants had to enter one paragraph of text (196 symbols) which represented daily conversation topics. This paragraph contained simple formulations of basic needs of a user (see an example of message in Figure 8). The experiment was performed with 2 subjects (both male, aged 24-28 years). The duration of experiment task was measured, and the input speed is presented in Figure 9.



I don't want to eat

Figure 8. Example of a message entered using the visual language. Source: The authors.

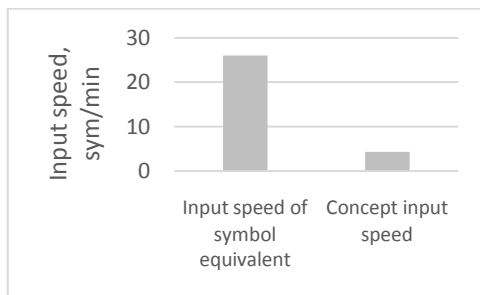


Figure 9. Input speed of concept-based speller. Source: The authors.

Two metrics of input speed are presented in Figure 10: concept input speed (number of valid concepts entered in a time unit) and input speed of letters in a textual language, which would be required to enter in a time unit in order to convey the same message. The results show that using the visual concept-based interface can increase input speed to 26.03 sym/min (equivalent to 4.31 concepts/min) as compared to 11.35 sym/min (see Table 2) using the traditional text-based interface of the EMG speller.

7 Evaluation and conclusion

We have described the development of the EMG speller application for an Assisted Living Environment based on a three-layered model of a Physiological Computing system based on the “operational modes” of Mann’s Humanistic Intelligence Framework. This system is controlled by voluntary muscular movements, particularly the orbicular ones (i.e., eye blinking), which are translated into text input commands.

The developed speller application is adaptive (input speed can be adapted dynamically in response to the user’s state) and intelligent (uses word complete and word frequency features). The application can aid people with reduced mobility by extending their autonomy in their home environment.

Two types of interfaces were developed and evaluated: traditional letter matrix-based interface and novel visual concept-based interface. The letter-matrix interface of the speller has been evaluated using accuracy, bit rate and input speed metrics. The achieved empirical results are within range of results achieved by other authors. The visual concept-based interface was evaluated using concept input speed and compared with equivalent text input speed. The results show an improvement of input speed by a factor of 2.3 as compared to best results achieved using a letter matrix based interface with a dictionary and adaptable input speed, which could be explained by conciseness of the visual language (both in terms of the size of dictionary and the length of visual „words“) as well as by reduced user effort required to communicate a message.

The advantages of using a visual language rather than a textual (alphabet-based) one are as follows: visual representation is easier to understand and communicate; common visual signs (such as computer icons) are universally known and understood; there is no need to implement multiple languages as the signs of a visual language would be understandable to all users; fewer grammar constraints; immediate visual feedback; smaller effort due to economy of concepts required for communication; input

speed is increased as the user has to select only one visual symbol (icon) instead of having to enter many different letters of a word.

The disadvantages of the visual language are ambiguity of interpretation of sentences consisting from visual symbols and limited expression capabilities.

We plan to address these problems in future by extending the dictionary of visual symbols implemented as well as by integrating an automatic translation plugin for conversion between the entered sequence of visual symbols and its textual counterpart.

Acknowledgement

The work described in this paper has been carried out within the framework the Operational Programme for the Development of Human Resources 2007-2013 of Lithuania „Strengthening of capacities of researchers and scientists“ project VP1-3.1-ŠMM-08-K-01-018 „Research and development of Internet technologies and their infrastructure for smart environments of things and services“ (2012-2015), funded by European Social Fund (ESF). The authors also would like to acknowledge the contribution of the COST Action IC1303 – Architectures, Algorithms and Platforms for Enhanced Living Environments (AAPELE).

References

- [1] Schroeder, A. and Wirsing, M. Developing Physiological Computing Systems: Challenges and Solutions. Proc. of Software Engineering Conference, LNI 198, GI 2012, pp. 21-36, 2012.
- [2] A. Marinc, C. Stockl w, A. Braun, C. Limberger, C. Hofmann, and A. Kuijper, Interactive personalization of ambient assisted living environments, Proc. of the 2011 Int. Conf. on Human interface and the Management of Information - Volume Part I (HI'11), LNCS vol. 6771, Springer-Verlag, Berlin, Heidelberg, pp. 567-576, 2011.
- [3] Allanson, J. and Fairclough, S.H. A research agenda for physio-logical computing. *Interacting with Computers*, 16(5), pp. 857-878, 2004.
- [4] Fairclough, S.F. Fundamentals of physiological computing. *Interacting with Computers*, 21(1-2), pp. 133-145, 2009.
- [5] Cecotti, H. Spelling with non-invasive Brain-Computer Interfaces – Current and future trends. *J. of Physiology-Paris*, 105 (1-3), 106-114, 2011.
- [6] Martisius, I. and Damasevicius, R., Class-Adaptive Denoising for EEG Data Classification, Proc. of 11th Int. Conference on Artificial Intelligence and Soft Computing, ICAISC 2012, Zakopane, Poland. LNCS vol. 7268, Springer, 302-309, 2012.
- [7] Martisius, I., Damasevicius, R., Jusas, V. and Birvinskas D. Using Higher Order Nonlinear Operators for SVM classification of EEG data. *Electronics and Electrical Engineering*, No. 3(119), pp. 99-102, 2012.
- [8] Speier, W., Fried, I. and Pouratian, N. Improved P300 speller performance using electrocorticography, spectral features, and natural language processing. *Clin. Neuroph.* 124(7), pp.1321-1328, 2013.
- [9] Birvinskas, D., Jusas, V., Martisius, I. and Damasevicius R. Data compression of EEG signals for artificial neural network classification. *Information Technology and Control*, 42(3), pp. 238-241, 2013.
- [10] Martisius, I., Birvinskas, D., Damasevicius, R., and Jusas V., EEG Dataset Reduction and Classification Using Wave Atom Transform, Proc. of 23rd International Conference on Artificial Neural Networks (ICANN2013), Sofia, Bulgaria, pp. 208-215, 2013.
- [11] Birvinskas, D., Jusas, V., Martisius, I. and Damasevicius R., EEG dataset reduction and feature extraction using Discrete Cosine Transform, Proc. of UK 6th European Modelling Symposium on Mathematical Modelling and Computer Simulation (EMS2012), pp. 199-204, 2012.
- [12] Martišius, I., Šidlauskas, K. and Damaševičius, R. Real-Time Training of Voted Perceptron for Classification of EEG Data. *International Journal of Artificial Intelligence (IJAI)*, 10 (S13), pp. 41-50, 2013.
- [13] Vasiljevas, M., Turčin, R. and Damaševičius R., EMG Speller with Adaptive Stimulus Rate and Dictionary Support, Proc. of FeDCSIS'2014: Federated Conference on Computer Science and Information Systems, Warsaw, Poland, 7 - 10 September, pp. 233-240, 2014.
- [14] Vasiljevas, M., Turčin, R. and Damasevicius R., Development of EMG-based speller, Proc. of INTERRACCION 2014: XV International Conference on Human Computer Interaction, Article No. 7, 2014.
- [15] Quek, M., H hne, J., Murray-Smith, R. and Tangermann, M., Designing Future BCIs: Beyond the Bit Rate, in Allison, B., Dunne, S., Leeb, R., Millan, J.D.R. and Nijholt, A. (eds.), *Towards Practical Brain-Computer Interfaces: Bridging the Gap from Research to Real-world Applications*, Springer, pp. 173-196, 2013.
- [16] Johnson, J. *Designing with the Mind in Mind: a Simple Guide to Understanding User Interface Design Rules*. Morgan Kaufmann, Burlington, 2011.
- [17] Mann, S. Wearable computing: Toward humanistic intelligence. *IEEE Intelligent Systems*, 16(3), pp. 10-15, 2001.
- [18] Lopes, J.B., Designing user interfaces for severely handicapped persons, Proc. of the 2001 EC/NSF Workshop on Universal accessibility of ubiquitous computing: providing for the elderly (WUAUC'01), ACM, New York, NY, USA, pp. 100-106, 2001.
- [19] Tomioka, R. and M ller, K.-R. A regularized discriminative framework for EEG analysis with application to brain-computer interface. *NeuroImage*, 49(1), pp. 415-432, 2010.
- [20] Lalitharatne, T.D., Teramoto, K., Hayashi, Y. and Kiguchi, K. Towards Hybrid EEG-EMG-Based Control Approaches to be Used in Bio-robotics Applications: Current Status, Challenges and Future Directions. *Journal of Behavioral Robotics*, 4(2), pp. 147-154, 2013.

- [21] Liu, Y., Zhou, Z., and Hu, D. Gaze independent brain-computer speller with covert visual search tasks. *Clinical Neurophysiology*, 122(6), pp. 1127-36, 2011.
- [22] Combaz, A., Chumerin, N., Manyakov, N.V., Robben, A., Suykens, J.A.K. and Van Hulle, M.M. Towards the detection of error-related potentials and its integration in the context of a P300 speller brain-computer interface. *Neurocomputing*, 80, pp. 73-82, 2012.
- [23] Hwang, H.-J., Lim, J.-H., Jung, Y.-J., Choi, H., Lee, S.W. and Im, C.H. Development of an SSVEP-based BCI spelling system adopting a QWERTY-style LED keyboard. *Journal of Neuroscience Methods*, 208(1), pp. 59-65, 2012.
- [24] Furdea, A., Halder, S., Krusienski, D.J., Bross, D., Nijboer, F., Birbaumer, N., Kübler, A. An auditory oddball (P300) spelling system for brain-computer interfaces. *Psychophysiology*, 46, pp. 617–25, 2009.
- [25] Brunner, P. and Schalk, G. Toward a gaze-independent matrix speller brain-computer interface. *Clinical Neurophysiology*, 122(6), pp. 1063-1064, 2011.
- [26] Ortner, R., Prueckl, R., Putz, V., Scharinger, J., Bruckner, M., Schnuerer, A. and Guger, C. Accuracy of a P300 Speller for Different Conditions: A Comparison. *Proc. of the 5th Int. Brain-Computer Interface Conference*, Graz, Austria, pp. 196, 2011.
- [27] Farwell, L.A. and Donchin, E. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalography and clinical neurophysiology*, 70(6), pp. 510-523, 1988.
- [28] Akram, F., Han, H.-S., Jeon, H.J., Park, K., Park, S.-H., Cho, J. and Kim, T.-S., An Efficient Words Typing P300-BCI System Using a Modified T9 Interface and Random Forest Classifier, *Proc. of 35th Ann. Int. Conf. of the IEEE EMBS*, pp. 2251-2254, 2013.
- [29] Townsend, G., LaPallo, B.K., Boulay, C.B., Krusienski, D.J., Frye, G.E., Hauser, C.K., Schwartz, N.E., Vaughan, T.M., Wolpaw, J. R. and Sellers, E.W. A novel p300-based brain-computer interface stimulus presentation paradigm: moving beyond rows and columns. *Clinical Neurophysiology*, 121(7), pp. 1109–1120, 2010.
- [30] Treder, M. and Blankertz, B. (C)overt attention and visual speller design in an ERP-based brain-computer interface. *Behavioral and Brain Functions*, 6, pp. 1–13, 2010.
- [31] Volosyak, I., Cecotti, H., Valbuena, D. and Gräser A., Evaluation of the Bremen SSVEP Based BCI in Real World Conditions, *Proc. of the 11th Int. Conference on Rehabilitation Robotics*, Kyoto, Japan, 322–331, 2009.
- [32] Acqualagna, L. and Blankertz, B. Gaze-independent BCI-spelling using rapid serial visual presentation (RSVP). *Clinical Neurophysiology*, 124(5), pp. 901-908, 2013.
- [33] Ulas, C. and Cetin, M., Incorporation of a language model into a brain computer interface based speller through HMMs, *Proc. of Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 1138-1142, 2013.
- [34] Mora-Cortes, A., Manyakov, N.V., Chumerin, N. and Van Hulle, M.M. Language Model Applications to Spelling with Brain-Computer Interfaces. *Sensors* 14, pp. 5967-5993, 2014.
- [35] Wolpaw, J.R., Birbaumer, N., Heetderks, W.J., McFarland, D.J., Peckham, P.H., Schalk, G., Donchin, E., Quatrano, L.A., Robinson, C.J. and Vaughan T.M. Brain-computer interface technology: A review of the first international meeting. *IEEE Transactions on Rehabilitation Engineering*, 8, pp. 164–173, 2000.
- [36] Pires, G., Nunes, U. and Castelo-Branco, M. Comparison of a row-column speller vs. a novel lateral single-character speller: Assessment of BCI for severe motor disabled patients. *Clinical Neurophysiology*, 123(6), pp. 1168-1181, 2012.
- [37] Käthner, I., Ruf, C.A., Pasqualotto, E., Braun, C., Birbaumer, N. and Halder, S. A portable auditory P300 brain-computer interface with directional cues. *Clinical Neurophysiology*, 124(2), pp. 327-338, 2013.
- [38] Shahriari, Y. and Erfanian, A. Improving the performance of P300-based brain-computer interface through subspace-based filtering. *Neurocomputing*, 121, pp. 434-441, 2013.
- [39] Vilic, A., Kjaer, T.W., Thomsen, C.E., Puthusserypady, S. and Sorensen, H.B., DTU BCI Speller: An SSVEP-based Spelling System with Dictionary Support, *35th Annual Int. Conf. of the IEEE EMBS*, pp. 2212-2215, 2013.
- [40] Wang, P.T., King, C.E., Do, A.H. and Nenadic, Z. Pushing the Communication Speed Limit of a Noninvasive BCI Speller, *CoRR abs/1212.0469*, 2012.

R. Damaševičius

Robertas Damaševičius graduated at the Faculty of Informatics, Kaunas University of Technology (KTU) in Kaunas, Lithuania in 1999, where he received a B.Sc. degree in Informatics. He finished his M.Sc. studies in 2001 (cum laude), and defended his Ph.D. thesis at KTU in 2005. Currently, he is a Professor at Software Engineering Department, KTU and lectures bioinformatics and robot programming courses. His research interests include brain-computer interface, bioinformatics, data mining and machine learning. He is the author or co-author of over 120 papers as well as a monograph published by Springer.

M. Vasiljevas

Mindaugas Vasiljevas graduated at the Faculty of Informatics, KTU in Kaunas, Lithuania in 2011. He received a B.Sc. in Informatics Engineering. He finished his M.Sc. studies at the same university in 2013. Currently he is a Ph.D. student at Kaunas University of Technology and works as an assistant at the same university. His Ph.D. research refers to physiological computing, neural computer interface and machine learning.

T. Šumskas

Tomas Šumskas graduated at KTU in 2013. He received BSc. degree in Informatics. Currently he is finishing his Masters studies in Software Engineering. His master project is related to human-computer interaction, and text autocorrection using machine learning algorithms