



KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
PROGRAMŲ INŽINERIJOS KATEDRA

AURELIJUS VAITKEVIČIUS

RANKOS GESTŲ ATPAŽINIMO, NAUDOJANT MAŠININIO
MOKYMO SI ALGORITMUS, TYRIMAS

Magistro darbas

Vadovas
doc. dr. T. Blažauskas

(data, parašas)

KAUNAS, 2016



KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
PROGRAMŲ INŽINERIJOS KATEDRA

AURELIJUS VAITKEVIČIUS

RANKOS GESTŲ ATPAŽINIMO, NAUDOJANT MAŠININIO
MOKYMO ALGORITMUS, TYRIMAS

Magistro darbas

Vadovas

doc. dr. T. Blažauskas

(data, parašas)

Recenzentė
doc. dr. Ingrida Lagzdinytė-
Budnikė

(data, parašas)

Studentas
A. Vaitkevičius

(data, parašas)

KAUNAS, 2016



KAUNO TECHNOLOGIJOS UNIVERSITETAS

Informatikos fakultetas

(Fakultetas)

Aurelijus Vaitkevičius

(Studento vardas, pavardė)

621E16001

(Studijų programos pavadinimas, kodas)

Baigiamojo projekto „Pavadinimas“

AKADEMINIO SAŽININGUMO DEKLARACIJA

20 ____ m. _____ d.
Kaunas

Patvirtinu, kad mano, **Aurelijaus Vaitkevičiaus**, baigiamasis projektas tema „Rankos gestų atpažinimo, naudojant mašininio mokymosi algoritmus, 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)

SANTRAUKA

Šiame darbe yra rašoma apie rankos gestų atpažinimą, naudojant mašinių mokymo algoritmus, bei „Leap Motion“ įrenginį. Šio įrenginio pagalba yra fiksuojama žmogaus ranka ir sudaromas jos erdvinis modelis. Gaunama visa reikalinga informacija apie ranką, tokia kaip pirštų padėtis erdvėje, jų vektoriai ir t. t. Šiame darbe pateiktą sistemą galima išmokyti atpažinti norimus rankos gestus, pateikiant rankos erdvinius modelius iš „Leap Motion“ įrenginio. Sistema paremta šablonų sulyginimo bei paslėptos klasifikacijos Markovo mokymosi algoritmais. Paslėptos klasifikacijos Markovo mokymosi algoritmas yra sukurtas atpažinti duomenų sekoms. Šis algoritmas visuomet gražina labiausiai atitinkantį gestą, dėl to sistema papildyta gestų filtravimo metodais, kurie nepateiks jokio rezultato, jei gestai yra nepanašūs. Atlikti tyrimai, kurie parodo kaip tiksliai sistema geba atpažinti skirtingus gestus bei kaip sudėtinga sistema mokyti juos atpažinti. Visi rezultatai, rastos problemos bei sprendimo būdai pateikti šiame dokumente.

SUMMARY

In this paper I analyzed gesture recognition using machine learning algorithms and Leap Motion device. Leap Motion generates a virtual 3D hand model by recognizing and tracking user hands. From this model Leap Motion application programming interface generates a lot of useful data like hand or finger location in 3D space, its vectors and etc. In this paper I presented a system that is capable of learning gestures by passing the data from the Leap Motion device to it. System uses template matching method and “Hidden Markov classification machine learning” (HMC) algorithm. The main use of HMC is to recognize similar data lists. This algorithm always returns a value, so few filtering methods were added to the system. Those methods will not return any value if the gesture is not similar. Experiments were conducted that show how fast the system algorithms can learn to recognize a gesture and how difficult it is to teach the system. All of the results are analyzed in this paper.

TURINYS

Paveikslėlių sąrašas	9
Lentelių sąrašas.....	10
TERMINŲ IR SANTRUMPŲ ŽODYNAS	11
1. Įvadas	12
1.1. Dokumento paskirtis.....	12
1.2. Darbo tikslas.....	13
1.3. Mokslinis naujumas	13
1.4. Uždaviniai	13
2. RANKOS GESTŲ ATPAŽINIMO SISTEMOS ANALIZĖ.....	14
2.1. Prie žmogaus rankos tvirtinami gestų atpažinimo įrenginiai	14
<i>2.1.1. Žmogaus rankos gestais paremtas roboto rankos valdymas</i>	<i>14</i>
<i>2.1.2. Gestų kalbos atpažinimas naudojant duomenų pirštines</i>	<i>15</i>
2.2. Objektų manipuliacijos virtualioje realybėje metodikos	16
2.3. Vaizdo atpažinimu grįstos gestų atpažinimo technologijos.....	18
<i>2.3.1. Microsoft Kinect</i>	<i>18</i>
<i>2.3.2. „Leap Motion“ įrenginys</i>	<i>19</i>
<i>2.3.2.1. „Leap Motion“ veikimo principas</i>	<i>19</i>
<i>2.3.2.2. „Leap Motion“ aplikacijų programavimo sąsaja.....</i>	<i>19</i>
<i>2.3.2.3. „Leap Motion“ tikslumo tyrimas</i>	<i>21</i>
<i>2.3.2.4. „Leap Motion“ teksto įvesties greičio tyrimas</i>	<i>21</i>
2.4. Gestų atpažinimas naudojant „Leap Motion“ įrenginį.....	23
2.1. Gestų atpažinimo algoritmai.....	25
<i>2.1.1. Paslėptasis Markovo modelis</i>	<i>25</i>
2.2. Skyriaus išvados	27
3. RANKOS GESTŲ ATPAŽINIMO SISTEMOS projektinė dalis	27
3.1. Sistemos paskirtis.....	27

3.2.	Sukurta gestų atpažinimo sistema	28
3.1.	Gestų atpažinimo sistemos tinklo servisas	28
3.2.	Galimi sistemos spartos tobulinimai	29
3.3.	Gestų šablonų saugojimas, bei filtravimas	30
3.4.	Mašinų mokymų algoritmų pritaikymas	30
3.5.	Gestų fiksavimas	31
3.6.	Funkciniai reikalavimai.....	34
3.7.	Nefunkciniai reikalavimai	34
3.7.1.	<i>Reikalavimai sistemos išvaizdai</i>	<i>34</i>
3.7.2.	<i>Reikalavimai sistemos išvaizdai</i>	<i>34</i>
3.7.3.	<i>Reikalavimai vykdymo charakteristikoms</i>	<i>34</i>
3.7.4.	<i>Reikalavimai veikimo sąlygoms.....</i>	<i>35</i>
3.7.5.	<i>Reikalavimai saugumui.....</i>	<i>35</i>
3.8.	Sistemos panaudos atvejų diagrama	36
3.9.	Sistemos architektūros modelis.....	42
3.9.1.	<i>Apžvalga</i>	<i>42</i>
3.9.2.	<i>Paketų detalizavimas</i>	<i>43</i>
3.10.	Naudojami trečiųjų šalių komponentai.....	48
3.11.	Naudojami aparatūrinė įranga	48
4.	RANKOS GESTŲ ATPAŽINIMO SISTEMOS TYRIMAI.....	49
4.1.	„Leap Motion“ sugeneruoto rankos modelio atvaizdavimo tikslumas	49
4.1.1.	<i>Amerikiečių gestų kalbos abėcėlės atpažinimas</i>	<i>51</i>
4.1.2.	<i>Lietuvių gestų kalbos abėcėlės atpažinimas</i>	<i>52</i>
4.2.	Sukurtos sistemos tyrimai, bei tolimesni tikslai	53
4.3.	Tyrimui reikalingas sistemos funkcionalumas	54
5.	RANKOS GESTŲ ATPAŽINIMO SISTEMOS EKSPERIMENTINĖ DALIS.....	55
5.1.	Mašinų algoritmų mokymas atpažinti lietuvių gestų kalbos abėcėlę raidžių gestus	

5.1.1. Tyrimo tikslas	55
5.1.2. Pasiruošimas tyrimui.....	55
5.1.3. Tyrimo eiga.....	56
5.1.4. Tyrimo rezultatai	57
5.2. Lietuvių kalbos gestų abėcėlės raidžių atpažinimo tyrimas.....	59
5.2.1. Tyrimo tikslas	59
5.2.1. Pasiruošimas tyrimui.....	59
5.2.2. Tyrimo eiga.....	59
5.2.3. Tyrimo rezultatai	60
5.3. Pangramos rašymas naudojant lietuvių gestų kalbos abėcėlės raidžių gestus	65
5.3.1. Tyrimo tikslas	65
5.3.1. Pasiruošimas tyrimui.....	65
5.3.1. Tyrimo eiga.....	65
5.3.2. Rezultatai	66
6. Išvados.....	69
7. Literatūra.....	70
8. Priedai	71

PAVEIKSLĖLIŲ SĄRAŠAS

1 pav. Roboto rankos valdymas naudojant gestų atpažinimą [7].....	15
2 pav. „SignAloud“ gestų atpažinimo pirštinės	16
3 pav. Leap Motion kaulų skeletas [2]	20
4 pav. Šeši laisvės kampai [23]	20
5 pav. Leap Motion pateikiamas rankos modelis trijų demencijų plokštumoje.....	24
6 pav. Paslėptojo Markovo modelio pavidys [1].....	26
7 pav. Sistemos išdėstymo diagrama	29
8 pav. Gestų fiksavimo posistemės būsenų diagrama	33
9 pav. Sistemos panaudos atvejų diagrama.....	36
10 pav. Sistemos paketų diagrama	42
11 pav. Paketo „Gesture Learning Algorithms“ klasių diagrama.	43
12 pav. Paketo "Gesture Recognition" klasių diagrama.	44
13 pav. Paketo „Leap Motion“ klasių diagrama.	44
14 pav. Paketo "WCF Service" klasių diagrama.....	45
15 pav. Paketo "Gestures Hand Data" klasių diagrama.	46
16 pav. Paketo "Gestures Data DB" klasių diagrama.	47
17 pav. Leap Motion įrenginys	48
18 pav. Lietuvių kalbos gestų abėcėlės raidžių gestai.....	50
19 pav. Amerikiečių gestų kalbos abėcėlės raidžių gestai	51
20 pav. Gesto atlikimo pavidys	56
21 pav. Lietuvių kalbos gestų abėcėlei atpažinti reikalingas šablonų skaičius.....	58
22 pav. Šablonų skaičiaus augimas	59
23 pav. Šablonų pokytis po tyrimu su pirmu dalyviu	60
24 pav. Kairėje netinkamas rankos erdvinis modelis. Dešinėje tinkamas.	61
24 pav. Šablonų pokytis po tyrimu su antru dalyviu.....	63
26 pav. Pirmojo ir antrojo dalyvio teksto įvedimo spartos suliginimas	68

LENTELIŲ SĄRAŠAS

Lentelė nr. 1. Leap Motion amerikiečių gestų kalbos raidžių atpažinimo duomenys	52
Lentelė nr. 2. „Leap Motion“ lietuvių gestų kalbos raidžių atpažinimo duomenys	52
Lentelė nr. 3. Gestų su judesiu šablonų skaičius	53
Lentelė nr. 4. Gestų be judesio šablonų skaičius	53
Lentelė nr. 5. Pirmojo tyrimo rezultatai. Gestui atpažinti reikalingas šablonų skaičius.	57
Lentelė nr. 6. Tyrimo su pirmu dalyviu rezultatai	62
Lentelė nr. 7. Tyrimo su antru dalyviu rezultatai	64
Lentelė nr. 8. Tyrimo su pirmu dalyviu rezultatai. Teksto įvedimo sparta.	66
Lentelė nr. 9. Tyrimo su antru dalyviu rezultatai. Teksto įvedimo sparta.....	67

TERMINŲ IR SANTRUMPŲ ŽODYNAS

VR – virtuali realybė

API (anlg. Application programming interface) – liet. Aplikacijų programavimo sąsaja

PS – programinė sistema

GDV – ant galvos dedamas vaizduoklis

3D – trijų dimensijų

REM – rankos erdvinis modelis

Gesto šablonas – atitinkamo rankos gesto erdvinio modelio išsaugoti duomenys, naudojami išmokyti gestų atpažinimo algoritmus.

1. ĮVADAS

Virtualios realybės įrenginių įdėja bandoma įgyvendinti jau seniai, tačiau tik dabar galime pastebėti didelį šuolį šių įrenginių kūrimo procese, dėl naujų produktų, tokių kaip „Oculus Rift“ [4] ar „Leap Motion“. „Oculus Rift“ – tai šalminis monitorius (anlg. Head-Mount Display), kuriame yra du atskiri vaizduokliai, po vieną kiekvienai akiai. Šie vaizduokliai transliuoja trijų dimensijų vaizdą vartotojui, taip jį įtraukdami į virtualią realybę. Virtualios realybės aplinkai valdyti neužtenka įprastų įvesties įrenginių tokių, kaip kompiuterinė pelė ar klaviatūra, kadangi vartotojui užsidėjus „Oculus Rift“ jis šių įrenginių nemato. Virtualios realybės aplinkai valdyti įprastai naudojami gestų atpažinimo įrenginiai, tokie kaip „Leap Motion“ [2]. „Leap Motion“ yra laikomas įvesties įtaisu, panašiai kaip kompiuterinė pelė ar jutiklinis ekranas, tačiau jo veikimas yra paremtas vaizdo atpažinimu. Naudojant dvi infraraudonųjų spindulių kameras, šis įrenginys gali atpažinti žmogaus rankas bei sukurti jų trijų dimensijų erdvinį modelį. Šis modelis gali būti naudojamas įvairioms vartotojo grafinėms sąsajoms valdyti. Sujungus „Oculus Rift“, bei „Leap Motion“ įrenginius gaunamas labai nuostabus produktas, kuris leidžia vartotojui matyt virtualų pasaulį bei sąveikauti su šio pasaulio elementais [5].

Nors „Leap Motion“ įrenginys geba atpažinti žmogaus rankas, tačiau jis nemoka atpažinti vartotojų rodomų gestų. Jis geba sumodeliuoti žmogaus ranką, bei sklandžiai pateikti jos duomenis trijų dimensijų lauke, tačiau funkcionalumo, kuris remdamasis šiais duomenimis galėtų pasakyti kada vartotojas rodo vieno ištiesto piršto gestą, šis įrenginys neturi. Žinoma yra programinės įrangos bibliotekų, kuriuose yra funkcijos, gebančios atpažinti kai kuriuos gestus, tokius kaip paspaudimas, sugriebimas [6], tačiau virtualios realybės aplinka reikalauja labai daug skirtingų gestų atpažinimo.

Darbe pateikiama sistema, kuri naudojant „Leap Motion“ įrenginį įrašinėje, kaip yra atliekami gestai ir šiuos duomenis pateikia kompiuterio mokymo algoritmams. Šie algoritmai mokosi iš šių duomenų ir vėliau geba atitinkamus gestus atpažinti. Sistemoje yra naudojamas „Paslėptos klasifikacijos Markovo mokymo algoritmas“ [1], kuris yra sukurtas atpažinti duomenų sekoms. Šios sistemos privalumas, jog norint išmokyti ją atpažinti gestus, nereikia nieko papildomai programuoti. Naudojant „Leap Motion“ įrenginį, gestas yra įrašinėjamas tol, kol jis yra atpažįstamas sistemos. Ši sistema gali būti pritaikoma virtualios realybės projektų kūrimui bei kituose projektuose, kuriuose reikalingas vartotojo rankos gestų atpažinimas. Vienas iš tokių projektų galėtų būti lietuvių kalbos gestų abėcėlės atpažinimas. Ši abėcėlė buvo naudojama šios sistemos tyrimui.

1.1. Dokumento paskirtis

Šiame dokumente pateikiama žmogaus rankos gestų atpažinimo, naudojant mašininio mokymo algoritmus, analizė, kurioje apžvelgiama taikomi metodai bei egzistuojančios sistemos. Taip pat pateikiama sukurtos rankos gestų mokymo bei atpažinimo sistemos projektas. Ši sistema remiasi

paslėptos klasifikacijos Markovo mokymo algoritmu, kuris sukurtas atpažinti duomenų sekoms bei šablonų sulyginimo metodu. Ataskaitoje pateikiami gauti rezultatai, taikant šiuos metodus žmogaus rankos gestų atpažinimui.

1.2. Darbo tikslas

Ištirti žmogaus rankos gestų atpažinimo tikslumą, naudojant mašininio mokymo algoritmus. Įvertinti ar naudojant „Leap Motion“ kaip įvesties įrenginį bei šiuos algoritmus, įmanoma sukurti įprastam vartotojui naudotis tinkantį produktą.

1.3. Mokslinis naujumas

1. Dokumentuotas rankos gestų atpažinimo metodas, naudojant „Leap Motion“ įrenginį bei mašininio mokymo algoritmus.
2. Suprojektuota ir sukurta rankos gestų atpažinimo sistema, kuri geba išmokyti naujus gestus.
3. Atlikti sistemos tyrimai, kurių tikslas išsiaiškinti, kaip tiksliai sistema gali atpažinti gestus, kaip sudėtinga išmokyti sistemą atpažinti šiuos gestus, kaip sparčiai vykdomas gestų fiksavimas ir atpažinimas.

1.4. Uždaviniai

1. Išanalizuoti gestų atpažinimo sistemas bei metodus.
2. Išanalizuoti mašininio mokymo algoritmus bei surasti, kuris metodas labiausiai tiktų atpažinti rankos gestus.
3. Išanalizuoti, kaip tiksliai „Leap Motion“ įrenginys geba sumodeliuoti trijų dimensijų rankos modelį.
4. Suprojektuoti ir sukurti rankos gestų atpažinimo sistemą.
5. Ištirti, kokių tikslumu galima atpažinti rankos gestus naudojant „Leap Motion“ įrenginį kartu su mašininio mokymo algoritmais.

2. RANKOS GESTŲ ATPAŽINIMO SISTEMOS ANALIZĖ

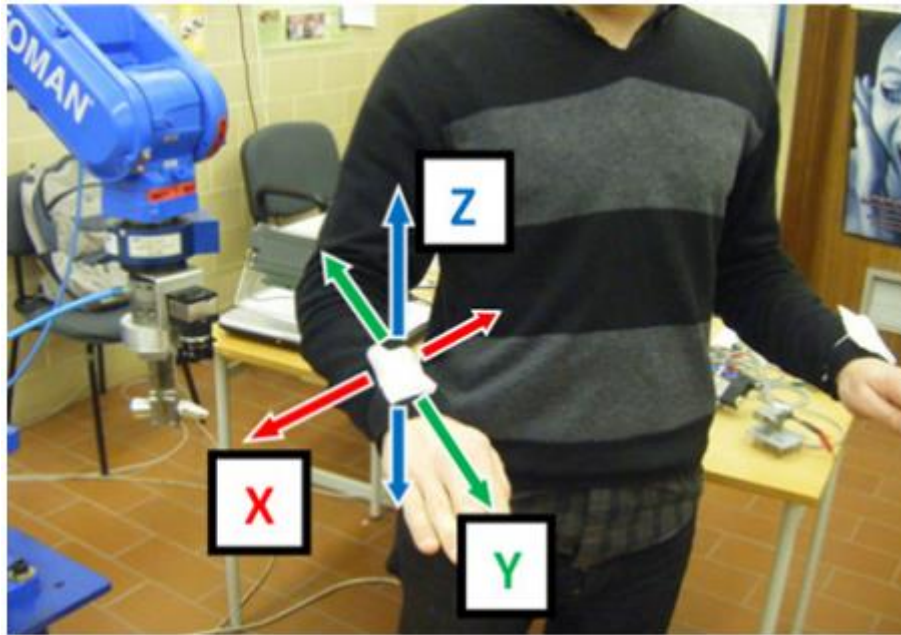
Kai kuri šiame darbe pateikta informacija buvo paimta iš kursinio darbo „Valdymo sąsajų skirtų navigavimui virtualioje erdvėje, naudojant virtualios realybės įrenginius, tyrimas“, kurį atlikau su kolega Mantu Taroza, programų sistemos inžinerijos magistro studijų studentu.

Rankos gestų atpažinimas yra plačiai vystomas, kadangi gali būti pritaikomas skirtingoms sritims. Labiausiai paplitę gestų atpažinimo įrenginiai gali būti skirstomi į prie žmogaus rankos tvirtinančius, arba grįstai vaizdo atpažinimu. Prie žmogaus rankos tvirtinantys gestų atpažinimo įrenginiai dažnai naudojami srityse, kurioms reikalingas labai didelis tikslumas, pvz. pramonėje valdant įrenginius. Vaizdo atpažinimu grįsti įrenginiai dažniausiai būna pritaikomi pramoginiams tikslais, kadangi jie yra patogesni vartotojui, jų pagaminimo kaštai yra mažesni, tačiau jie pasižymi prastesniu tikslumu.

2.1. Prie žmogaus rankos tvirtinami gestų atpažinimo įrenginiai

2.1.1. Žmogaus rankos gestais paremtas roboto rankos valdymas

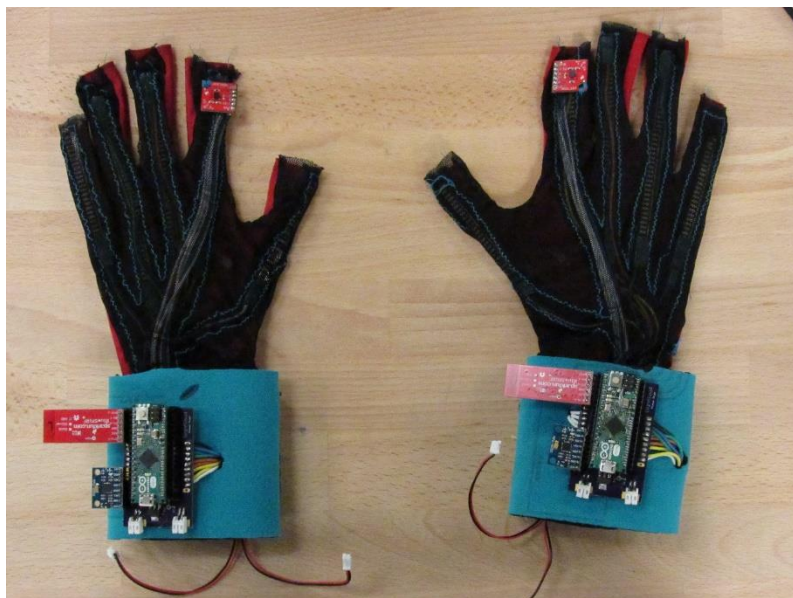
Pramonėje gestų atpažindinimas gali būti naudojamas įrenginiams valdyti, tokiems kaip roboto ranka [7]. Naudojant ekselerometro technologiją, kuri paprastai būna pritvirtinta prie riešo, galima nuskaityti vartotojo rodomus gestus. Naudojant mašininio mokymo algoritmus, sistema išmokoma tam tikrus vartotojo judesius atpažinti. Mokslininkai Pedro Neto, J. Norberto Pires, ir A. Paulo Moreira, šį veikimo principą pritaikė roboto rankai valdyti. Roboto ranka atkartodavo rankos judesį trijų dimensijų erdvėje, t.y roboto ranka galėjo judėti pirmyn, atgal, į šonus, bei į viršų ar apačią (*1 pav.*). Jų tyrimo rezultatai parodė, jog prasčiausiai atpažįstamas gestas veikia 82 % tikslumo. Kiti gestai viršijo 90 %, bei netgi siekė 100 %. Ištaisius kelis jų paminėtus trūkumus, sistema gali būti naudojama įvairiems įrenginiams valdyti.



1 Pav. Roboto rankos valdymas naudojant gestų atpažinimą [7]

2.1.2. Gestų kalbos atpažinimas naudojant duomenų pirštines

2016 m. labai pagarsėjo Thomas Pryor ir Navid Azodi Vašingtono Universiteto studentų išradimas pavadinimu „SignAloud“. „SignAloud“ įrenginys yra pirštinės kurios gali atpažinti amerikiečių gestų kalbos rodomus gestus (2 pav.). Pirštinėse įmontuoti sensoriai panašiai kaip roboto rankos valdymo sistemoje seka vartotojo rankos judesius. Šioje sistemoje yra sekami netik riešo padėtis erdvėje, bet ir pirštų. Iš pirštinių gautą informaciją yra siunčiama serveriui kuris įvairias regresijos ir klasifikavimo algoritmais, panašiai į neuroninius tinklus, šiuos gestus atpažįsta, bei garsu pateikia atpažintą žodį. Šis išradimas 2016 metų pavasarį laimėjo prestižinio Lemelson-MIT konkurso dešimties tūkstančių dolerių prizą. Kadangi šis įrenginys yra naujas, bei dar neišleistas į rinką yra nežinomas jo tikslumas, tačiau sprendžiant pagal pasirinktą aparatūrinę įrangą, bei sistemos algoritmus, šis įrenginys yra daug žadantis.



2 pav. „SignAloud“ gestų atpažinimo pirštinės

2.2. Objektų manipuliacijos virtualioje realybėje metodikos

Gestų atpažinimo sistemos yra dažnai naudojamos virtualios realybės sistemose. Vienas naujesnių virtualios realybės atvaizdavimo įrenginių yra „Oculus Rift“. Šis ant galvos dedamas vaizduoklis nuolat skaičiuoja vartotojo galvos poziciją ir atnaujina vaizdą, suteikdama vartotojui natūralų būdą stebėti virtualų pasaulį. Problema kyla kai vartotojas užsidėjęs ant galvos šį įrenginį nemato aplinkinio realaus vaizdo, todėl jam sunku naudotis įprastai vartotojo sąsajos įrenginiais, tokiais kaip kompiuterinė pelė ar klaviatūra. Taip pat jis virtualiame pasaulyje nemato savo rankų, todėl virtualios realybės iliuzija išblėsta. Suderinus „Oculus Rift“ su gestų atpažinimo technologija, vartotojas gali matyti savo rankas bei sąveikauti su virtualios realybės objektais.

Vienas svarbiausių dalykų kalbant apie virtualią realybę (VR) yra vartotojo sąveika su pasauliu. Šią sąveiką su VR galima apibrėžti kaip objektų, vaizduojamų VR, manipuliaciją. Tam yra įvairių technologijų, iš kurių populiariausios remiasi rankų gestais. Mokslininkų grupė iš Georgijos Technologijų Instituto (Atlanta, JAV) siekė palyginti šešias technologijas, skirtas objektų virtualioje realybėje manipuliacijai.

Tiriamos yra „Go go“, „Fast go go“, „Strech go go“, „Indirect stretching“, „Ray casting“ ir „Ray casting with reeling“.

„Go go“ – tai metodika, kai vartotojas valdo virtualias rankas kaip atitikmenį realioms rankoms. Arti esantį objektą patogiu tiek paimti, tiek jį sukiooti ir padėti į kitą vietą. Problema kyla, jeigu norimas manipuluoti objektas yra nepasiekiamas rankoms. Dėl to „Go go“ turi tam tikrą ribą nutolusią nuo vartotojo, kurią peržengus rankoms, virtualios rankos pailgėja daugiau, negu realios rankos atitolsta

nuo vartotojo. Kitais žodžiais tariant, susiejant realios rankos ir virtualios rankos ilgus funkcija, iki ribos funkcija yra linijinė, o už ribos yra ne linijinė, auganti funkcija. Nors tai padeda pasiekti toliau esančius objektus, juos pasidaro sunku manipuluoti kai yra taip atitolę, be to, objekto sugriebimas pasidaro sunkesnis.

„Fast go go“ iš esmės yra ta pati „Go go“ metodika, tik skiriasi tuo, jog nėra 3D erdvės, kurioje virtualios rankos atitiktų realias savo ilgiu, t. y. funkcija neturi ribos, nuo kurios ji nėra linijinė. Virtuali ranka progresuojančiai ilgėja nuo pat vartotojo (kaip atskaitos taško). Ši metodika leidžia pasiekti dar toliau esančius objektus, tačiau vis dar kenčia nuo tų pačių problemų, kaip sudėtingas tolimų objektų sugriebimas.

„Strech go go“ yra modifikuota „Go go“ metodika tuo, jog išėjimas iš ribų lemia ne tiesioginę virtualios rankos ilgio priklausomybę nuo realios rankos, bet virtualios rankos ilgėjimo greitį. Ši metodą apibrėžia trys zonos, esančios priešais vartotoją. Kol rankos yra vidurinėje zonoje, virtualios rankos tiesiogiai atkartoja realias rankas. Su rankomis išėjus už zonos ribų į trečią, labiausiai nutolusią zoną, rankos pradeda ilgėti, ir kuo labiau rankos nutolusios, tuo greičiau jos ilgėja. Ilgėjimo procesas sustabdomas realioms rankoms grįžus į antrąją zoną. Jei rankos patenka į pirmąją zoną, virtualios rankos yra trumpinamos. Ši metodika realiai leidžia pasiekti bet koku atstumu nutolusius objektus, nes virtualių rankų ilgis tampa valdomas neribotai.

„Indirect stretching“ – metodas, panašus į „Go go“ metodą, tik virtualios rankos ilgis valdomas ne gestais, o naudojant 3D pelę. Šis metodas yra labai panašus į „Strech go go“ tuo, jog nėra ribojamas maksimalus virtualių rankų ilgis, tačiau turi ženklų privalumą: paprasčiau išlaikyti ilgį pastovų ir taip pagriebti objektą ir jį valdyti.

„Ray casting“ – tai metodas, kai yra virtualioje realybėje yra naudojami ne žmogaus rankų virtualios imitacijos, bet spindulys. Spindulys leidžia labai tikslų objektų pagriebimą, nes galima tiesiogiai nusitaikyti į objektą nekreipiant dėmesio į atstumą iki jo, tačiau manipuliacija objektu yra taip smarkiai apribota, kad iš esmės jį galima tik suklikti pagal spindulio ašį.

„Ray casting with reeling“ – šis metodas iš esmės sprendžia „Ray casting“ metodo problemą dėl ribotos objektų manipuliacijos. Prie spindulio pridedama galimybė naudojant trijų dimensijų pelę objektą pritraukti prie vartotojo arba atitolinti pagal spindulį, kurio pagalba objektas yra manipuluojamas.

Eksperimentinio tyrimo metu, vienuolikai savanorių buvo leidžiama laisvai išbandyti visas metodikas virtualioje realybėje objektų manipuliacijai atlikti. Buvo nustatyta, jog „Go go“ metodikos yra patogios manipuluoti objektais, tačiau prastos, kai reikėdavo objektus pagriebti per atstumą,

didesnį, negu rankos ilgis. Tuo tarpu „Ray casting“ vienbalsiai buvo pastebėta, jog yra ypač patogus objektų pagriebimui, tačiau visiškai netinkamas jų manipuliacijai. Dėl šių priežasčių originali problematika, t.y. žmogaus sąsaja su virtualios realybės objektais, buvo išskirta į dvi atskiras: objektų pagriebimas ir jų manipuliacija. Atsižvelgiant į tai, jog pirmą problemą sprendžia „Ray casting“, o antrąją „Go go“ metodai, buvo sukurtas hibridinis metodas „HOMER“ (Hand-centered Object Manipulation Extending Ray-casting).

„HOMER“ metodas susideda iš to, jog pradžioje objekto pagriebimas įvykdomas su spinduliu („Ray casting“), o po to objekto manipuliacija vykdoma rankų gestų pagalba ar naudojant 3D pelę.

Tyrimo pabaigoje pateikiama išvada, jog sąveikoje su VR objektais svarbu atskirti objektų pagriebimo ir manipuliacijos uždavinius ir jog tuo pagrindu sukurtas „HOMER“ metodas parodė, jog yra iš tiesų patogus metodas VR sąsajos vystymui.

2.3. Vaizdo atpažinimu grįstos gestų atpažinimo technologijos

2.3.1. Microsoft Kinect

Vienas iš populiariausių šių dienų žmogaus sąsajos su kompiuterių įrenginių yra kompanijos „Microsoft“ 2010 metų lapkritį išleistas „Kinect“. Šis įrenginys sukuria natūralią vartotojo sąsają, leisdamas vartotojui valdyti programas ar žaidimus fiziškai neliečiant jokio įrenginio[9]. Kinect tai realizuoja sekdamas vartotojo kūno judesius, gestus ir balso komandas[12,13]. Kinect naudoja Izraelio kompanijos „PrimeSense“ sukurtą technologiją, kuri generuoja kambario gylio, spalvų ir garsų duomenis realiu laiku. Jis gali veikti bet kokiame kambaryje, prie įvairių apšvietimo sąlygų, nesvarbu ar tai būtų visiška tamsa, ar pilnai apšviestas kambarys[14,15].

Kinect sudaro horizontalus vamzdis, pritvirtintas prie platformos su motoru (kad galėtų sekti vartotoją) ir yra sukurtas taip, kad galėtų būti patalpintas tiek virš, tiek po televizoriaus ekranu. Įrenginys turi RGB(angl. Red/Green/Blue, liet. raudona/žalia/mėlyna) spalvų kamerą, gylio sensorių (sudarytas iš infraraudonųjų spindulių projektoriaus ir kameros) ir triukšmą šalinančių mikrofonų (keturi mikrofonai, kurie taip pat gali būti panaudoti fiksuoti žmogaus pozicijos trijose dimensijose)[46,47]. „Kinect“ gali pateikti pilną 3D žmogaus kūno vaizdą, atpažinti gestus, veidą ir balsą[13,14].

„Kinect“ įrenginio pritaikymo sritys apima ne tik kompiuterinius žaidimus ir pramogas[16] bet jau pradamas taikyti ir tokiose srityse kaip:

- 3-D ir patobulintos vaizdo konferencijos. Tyrimai atliekami Oliver Kreylos California Davis universitete naudojant du „Kinect“ įrenginius[21,22], bei „Kinected Conference“

tyrimas, atliekamas Lining Yao, Anthony DeVincenzi ir kitų MIT Medios laboratorijoje[17,18],

- „Kinect“ naudojimas medicinoje atpažinti protines vaikų negalias. Tyrimas vykdomas Nikolaos Papanikolopulos ir kitų Minesotos universitete[19],
- „Virtopsy“ projekte „Kinect“ naudojamas kaip medicinos įrenginių kontrolės prietaisas operacijos metu, tiesiogiai neliečiant medicininių instrumentų. Projektas vykdomas Forensic Medicine institute, Bern universitete, Šveicarijoje[20].

2.3.2. „Leap Motion“ įrenginys

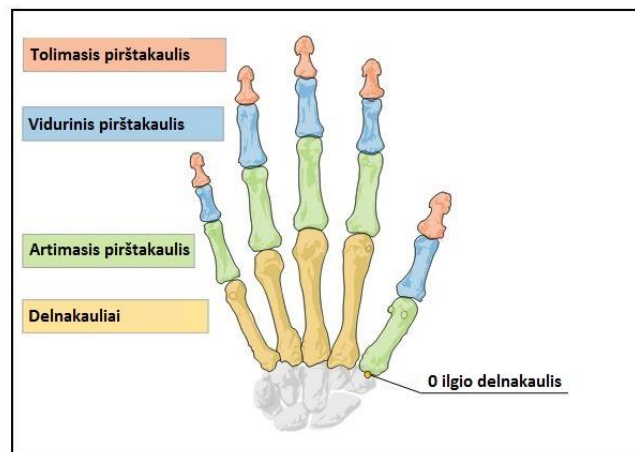
2.3.2.1. „Leap Motion“ veikimo principas

„Leap Motion“ turi dvi monochromines infraraudonųjų spindulių (IS) kameras ir tris IS šviesos diodus. Šiais šviesos diodais yra sugeneruojama 3D IS taškų modelis, kuris yra užregistruojamas monochrominėmis kameromis. „Leap Motion“ technologija yra išpūdinga ne aparatūriniu požiūriu, bet programine įranga. Užregistruotus IS modelius „Leap Motion“ išanalizuoja matematiniais algoritmais, kurių kompanija iki šiol neatskleidė. Iš dviejų 2D vaizdų, gautų monochrominėmis IS kameromis, „Leap Motion“ sugeneruoja rankų erdvinį modelį. Priešingai nei Microsoft kompanijos kuriamas „Kinect“, kuris traktuoja pilną žmogaus skeletą, „Leap Motion“ seka tik vartotojų rankas. Įvertinant žmogaus rankos galimybes, „Leap Motion“ programinė įranga gali numanyti pirštų, delno ar riešo padėtis ir tuo atveju, kai kameros šių vietų nemato. Rankas „Leap Motion“ gali traktuoti nuo 25 iki 600 milimetrų atstumu, 150 laipsnių kampu, leidžiant vartotojui laisvai jaustis erdvėje. Nors ir šios charakteristikos yra svarbios, pats didžiausias „Leap Motion“ privalumas yra jo programavimo sąsaja (PS), kuris leidžia lengvai kurti taikomąją programinę įrangą.

2.3.2.2. „Leap Motion“ aplikacijų programavimo sąsaja

Naudojant infraraudonųjų spindulių kameras, galima nustatyti kiekvieno rankos taško koordinates. Norint atpažinti rankų gestus reikia apdoroti labai daug duomenų nustatant rankos dalis (riešą, plaštaką, pirštus). „Leap Motion“ privalumas yra tas, jog šios sudėtingos duomenų analizės funkcijos yra įdiegtos į „Leap Motion“ programinį paketą [2]. Šis paketas naudingas tuo, jog jis iš 3D vaizdo gauna rankų erdvinį skeletą, išanalizuoja ir sudeda į tam tikrus objektus, kuriuose saugoma atitinkama rankos dalies informacija. „Leap Motion“ PS turi tris pagrindinius rankos objektus: pilna ranka, delnas, pirštai. Pilnos rankos objektas suteikia informaciją apie rankos poziciją erdvėje, jos ilgį, bei plotį. Delno objekte saugoma informacija apie rankos identiškumą (kairė ar dešinė), poziciją, bei tos rankos pirštų sąrašas. Pati svarbiausia „Leap Motion“ PS dalis, reikalinga gestų traktavimui yra pirštų objektas. Šiame objekte saugoma kiekvieno žmogaus rankos piršto pagrindiniai kaulų duomenys. Žmogaus rankos pirštas turi keturis pagrindinius kaulus (jei įskaitysime plaštakos dalį),

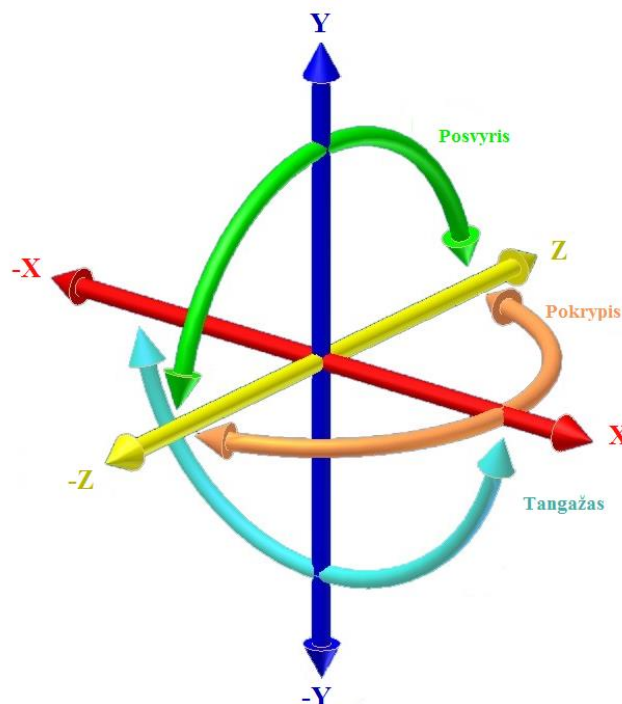
išskyrus nykštį, kuris turi tik tris. „Leap Motion“ PS piršto objekte nykštys kaip ir kiti pirštų objektai turi keturis kaulus, tačiau apie vieną iš jų niekad nepateikiama jokia informacija (3 pav.).



3 pav. Leap Motion kaulų skeletas [2]

Padėtis erdvėje „Leap Motion“ nusako šešiais laisvės laipsniais (4 pav.).

- X, Y, Z – trimatės koordinačių sistemos ašys.
- Tangažas - kampinis posūkis vertikalioje plokštumoje (Y ašies)
- Pokrypis - kampinis posūkis horizontalioje plokštumoje (X ašies).
- Posvyris – kampinis posūkis Z ašies.



4 pav. Šeši laisvės kampai [23]

2.3.2.3. „Leap Motion“ tikslumo tyrimas

Įrenginių naudojimas virtualioje realybėje taip pat kelia klausimą, koks yra jų tikslumas. Akivaizdu, jog patogiausi tie įrenginiai, kurių jautrumas yra labai didelis ir fiksuoja net mažiausius judesius. Mokslininkai iš Dortmundo Technologijos Universiteto (Technical University Dortmund, Germany) siekė pirmieji išsiaiškinti, koks yra tikras įrenginio „Leap Motion“ jautrumas, fiksuojant judesius [24].

Tyrimo atlikimui buvo pasirinktas robotas „Kuka Robot KR 125/3“, kurio judesių tikslumas yra mažiau nei 0,2 mm. Žmogus šiam tyrimui netinka dėl to, jog raumenys turi nevalingus ritminius raumenų judesius, kurių amplitudės gali siekti 0,2 – 0,6 mm jauniems ir 0,5 – 1,7 mm vyresnio amžiaus individams [25,26]. Tuo tarpu, „Leap Motion“ gamintojai skelbia, jog jautrumas fiksuojant pirštų galiukus yra apie 0,01 mm.

Testai vykdomi dviejų tipų:

1. Statinis – siekiama sulygtinti pozicijas, kurias pateikia individualiai roboto valdymo įrankiai ir „Leap Motion“;
2. Dinaminis – erdvėje piešiamos linijos ir lyginamos pozicijos, fiksuojamos „Leap Motion“ ir roboto kontrolės įrankio.

Tyrimas atliekamas fiksuojant į roboto ranką įmontuotą pieštuką laikant kaip atskaitos tašką. Atlikus tyrimą, buvo gauti tokie rezultatai:

- Statinės analizės metu, 3D pozicijos tikslumas (skirtumas tarp „Leap Motion“ ir roboto rodmenų) buvo gautas mažesnis negu 0,2 mm; taip pat naudojant skirtingo dydžio pieštukus, nebuvo nustatyta įtaka pozicijos tikslumo matavimui;
- Dinaminės analizės metu buvo pasiektas didžiausias 2,5 mm tikslumas, vidutiniškai 1,2 mm.

Taigi, tyrimo dėka, galima teigti, jog realiomis praktinėmis sąlygomis, skelbiamo tikslumo negalima pasiekti. Tačiau, „Leap Motion“ pasiekiamas tikslumas vis tiek lenkia kitus gestų valdomas kompiuterio sąsajas, tokias kaip Microsoft Kinect [24].

2.3.2.4. „Leap Motion“ teksto įvesties greičio tyrimas

Kauno Technologijos Universiteto Programų Inžinerijos katedroje analizuota problema susijusi su vartotojo sąsajų kūrimu, teksto įvedimo būdais, bei jų patogumu [30]. Pagrindinis tyrimo tikslas, išsiaiškinti senų, bei naujų įrenginių patogumą bei našumą naudojantis. Dažniausiai naudojami įvesties įrenginiai prieinami paprastam vartotojui yra:

- Pelė;
- Klaviatūra;

- Pultelis;
- Liečiamas ekranas;
- Mikrofonas
- Kamera;
- Judesių atpažinimo sensorius;

Tyrimui pasirinkti šie vartotojo sąsajų įrenginiai:

- Televizoriaus pultelis;
- „Leap Motion“;
- Klaviatūra.

Tyrimo metu vertinti įrenginių trūkumai, privalumai ir teksto įvedimo greitis. Siekta nustatyti įrenginių pritaikomumą išmaniajam televizoriui.

Analizuojant šiuos įrenginius, nustatytos jų stiprybės:

- Pultelis: neužima daug vietos, turi daug televizoriaus valdymo funkcijų;
- „Leap Motion“: neapribotas atstumas, greitas teksto įvedimas, neužima daug vietos, gestai – natūralus žmogaus elgesys;
- Klaviatūra : labai greitas teksto įvedimas.

Taip pat buvo paminėti jų trūkumai:

- Pultelis: ilga įvesties trukmė, didelis klaidų skaičius;
- „Leap Motion“: varginanti įvestis, didelis klaidų skaičius, reikalingas serveris, stabilus nejudantis paviršius;
- Klaviatūra: apribotas atstumas, užima daug vietos.

Teksto įvedimo greičio bandymo metu įvedamas 29 simbolių tekstas (svetainės adresas) be teksto suflerio. Pirmojo bandymo rezultatai:

- „Leap Motion“ – 43.5s;
- Pultelis – 91.5s;
- Klaviatūra – 6s.

Paskutiniojo, penktojo, teksto įvedimo bandymo rezultatai:

- „Leap Motion“ – 29.5s;
- Pultelis – 69.5s;
- Klaviatūra – 6s.

Šie rezultatai atskleidžia, kokius pranašumas turi „Leap Motion“ įrenginys, naudojantis žmogaus gestų atpažinimo technologiją, lyginant su televizoriaus pulteliu, bet vis tiek nusileidžia klaviatūrai, kuri išlieka sąsaja, leidžianti tekstą įvesti greičiausiai. Taip pat pastebėta, kad kiekvieno sekančio bandymo metu teksto įvedimo trukmė sumažėja. Tai kartu parodo ir tai, jog įvedimo trukmė tiesiogiai

priklauso nuo vartotojo įgūdžių naudotis pateikiama įranga ir kuo daugiau turima patirties, tuo greitesnis teksto įvedimas.

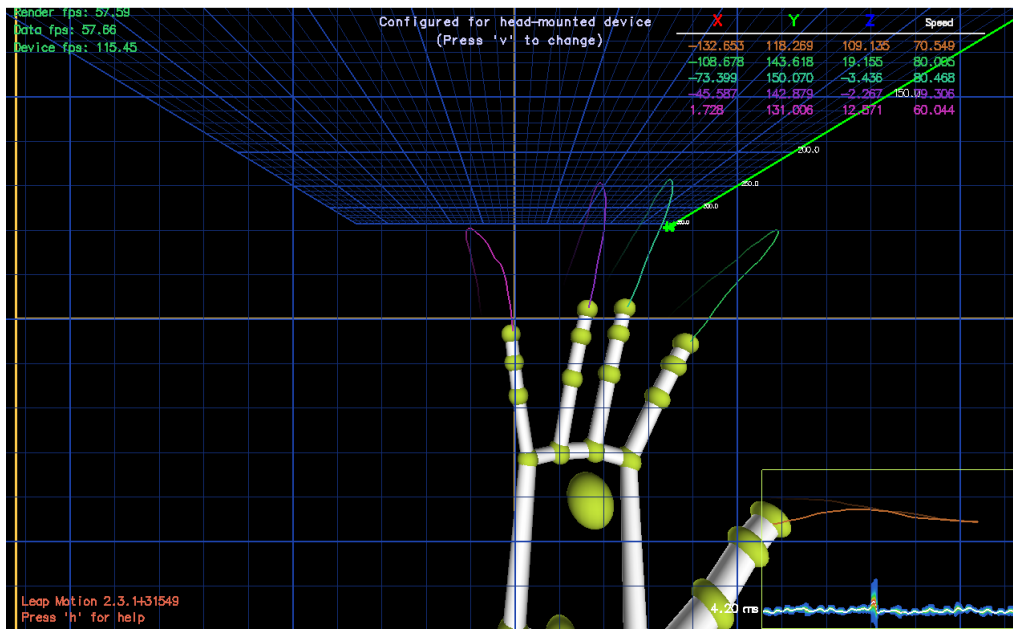
Po „Leap Motion“ įrenginio bandymų, vartotojai pateikė keletą būdų, kaip būtų galima patobulinti sąsają su šiuo įrenginiu:

- Žymeklio judesių jautrumo mažinimas, artėjant paspaudimo zonai;
- Vaizdo priartinimas teksto įvedimo metu;
- Žodžių statistinis tikrinimas iš žodyno;
- Kelių pirštų atpažinimas vienu metu.

2.4. Gestų atpažinimas naudojant „Leap Motion“ įrenginį

Nors „Leap Motion“ įrenginys geba atpažinti žmogaus rankas, tačiau jis nemoka atpažinti vartotojų rodomus gestus. Jis geba sumodeliuoti žmogaus rankos erdvinį modelį, tačiau funkcionalumo kuris remiantis šiais duomenimis galėtų pasakyti kada vartotojas rodo vieno ištiesto piršto gestą šis įrenginys neturi. Žinoma yra programinės įrangos bibliotekų, kuriuose yra funkcijos gebančios atpažinti kai kuriuos gestus, tokius kaip paspaudimas, sugriebimas, tačiau virtualios realybės aplinka reikalauja labai daug skirtingų gestų atpažinimo. Atskirai įdiegti į sistemą kiekvieno reikalingo gesto atpažinimą kainuotų labai daug laiko.

Atpažinti skirtingus gestus yra ir lengvesnis būdas kurį aptarsime šiame darbe. „Leap Motion“ įtaisas, kaip jau minėjau pateikia žmogaus rankos erdvinį modelį trijų dimensijų erdvėje. Šio modelio pagalba galima gauti kiekvieno rankos piršto, jų kaulų ar delno centro koordinatas, posūkio kampus, bei kitos reikalingos informacijos. Jei šis įtaisas bus pastatytas visuomet toje pačioje pozicijoje priešais vartotoją, ir jam bus rodomas tas pats gestas, jis pateiks beveik tuos pačius duomenis, t. y. su labai maža paklaida.



5 pav. Leap Motion pateikiamas rankos modelis trijų demencijų plokštumoje

Vadinasi į kiekvieną rankos gestą, kuris neatlieka jokio judesio, galima žiūrėti kaip į rankos, bei jos kiekvieno piršto padėtį erdvėje. Jei mes rodytumo nejudantį gestą, bei įrašytumėme iš „Leap Motion“ gaunamus duomenis į duomenų bazę, vėliau juos galėtumėme naudoti kaip šabloną atpažinti tam gestui.

Gestų, kurie neatlieka judesį, atpažinimui reikalingi tik trys laisvės laipsniai: posvyrio, pokrypio ir tangažo. Trimatės koordinatinių sistemos ašys X, Z, Y yra nesvarbios kadangi vartotojas gali rodyti gestą, bet kurioje sistemos vietoje. Jei rankos, bei pirštų posvyrio kampai yra panašūs, rodomas gestas yra atpažįstamas. Nenaudojant ašių X, Y, Z taip pat išsprendžiama nevienodų, skirtingų vartotojų pirštų problema.

Gestų atpažinimas kurie atlieką tam tikrą judesį vyksta labai panašiai. Duomenų bazėje saugoma jau nebe vieno rankos erdvinio modelio duomenis, bet kelių. Vidutiniškai galima imti nuo penkių iki dešimties kadrų per sekundę. Rodant gestą su judesiu algoritmas tikrina kaip kiekvieno kadro duomenys kinta. Jei kitimas yra panašus su duomenų bazėje esančiais duomenimis, gestas yra atpažįstamas. Šiam atpažinimui reikalinga visi šeši laisvės kampai. X, Y, Z ašių duomenys yra paslenkami link nulio, kadangi nereikia tikrinti nuo kurio taško vartotojas pradėjo rodyti gestą, bet kaip šių ašių duomenys kito rodant gestą.

Šis atpažinimo būdas yra universalesnis, kadangi sistema gebėtų pati išmokti kaip gestas yra atliekamas. Nereikėtų į sistemą įdiegti papildomos logikas norint atpažinti naujus gestus. Tačiau šis šabloninis gestų atpažinimas turi ir keletą problemų:

Yra sunku surasti toleruotiną paklaidą tarp rodomo rankos gesto erdvinio modelio, bei duomenų išsaugotų duomenų bazėje. Ši paklaida taip pat gali būti skirtinga kiekvienam pirštui. Sunku yra nustatyti kuriuos kadrus su kuriais lyginti, kadangi gestas gali būti rodomas skirtingu greičiu. Kadru suliginimas kainuotų labai daug resursų.

Kompiuterio mokymo algoritmai šiai problemai išspręsti turėtų labai pagelbėti. Šie algoritmai yra unikalūs, kadangi leidžia kompiuteriui įdiegti tam tikrą funkcionalumą pateikiant labai daug duomenų ar kitaip tariant tam tikros sąlygos pavyzdžių. Su „Leap Motion“ įrenginių įrašyti šablonai yra labai geri duomenys šiems algoritmams mokytis. Atpažinti duomenų sekas yra sukurtas mašinių mokymo algoritmas pavadinimu „Paslėptos klasifikacijos Markovo mokymo algoritmas“. Šis algoritmas turėtų padėti išspręsti nukrypimo nuo tam tikros šablonų sekos problemą.

2.1. Gestų atpažinimo algoritmai

2.1.1. Paslėptasis Markovo modelis

Laibai paprastu pavyzdžiui šiame skyriuje paaiškinsime, kaip veikia paslėptasis Markovo modelis[1], bei kaip jis gali būti pritaikytas gestų atpažinimui. Įsivaizduokime jog turime du gestus: A, ir B. Abu šiuos gestus galime nusakyti tam tikromis reikšmėmis: X1, X2, X3.

Tikimybė jog gesto A reikšmės yra:

- $X1 = 0.1$
- $X2 = 0.4$
- $X3 = 0.5$

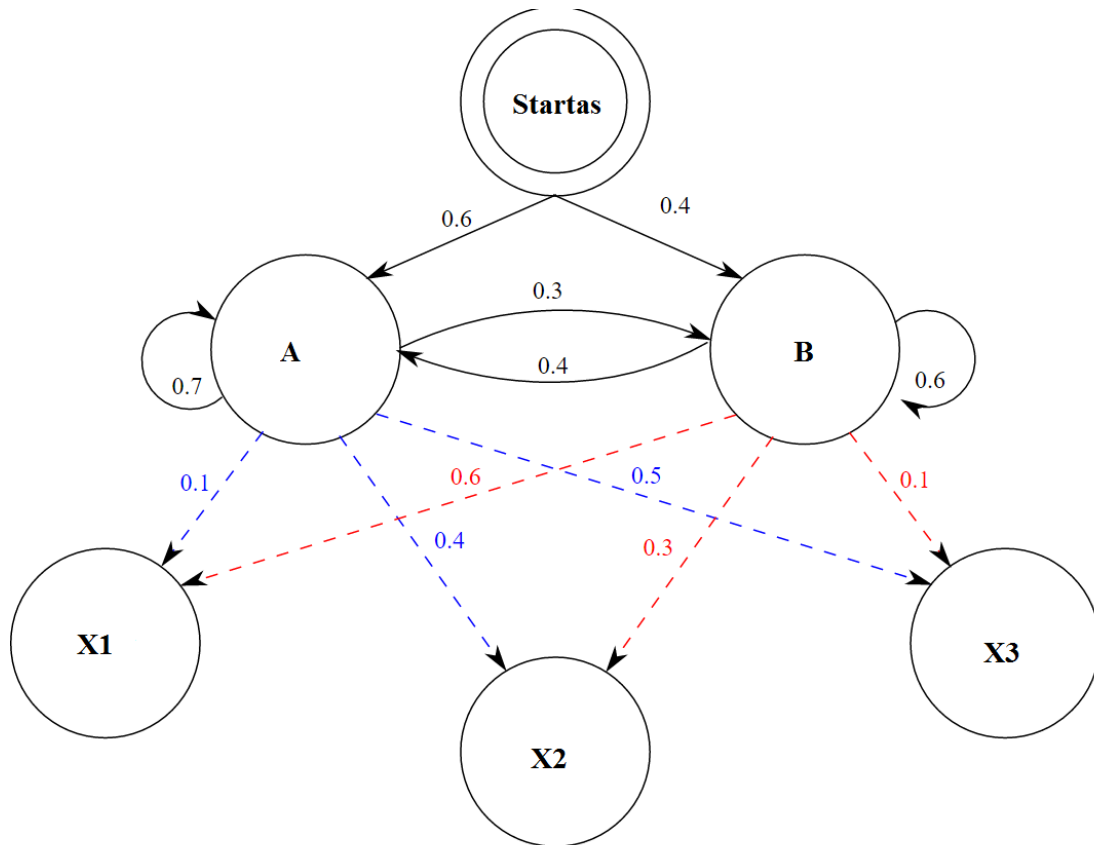
Tikimybė jog gesto B reikšmės yra:

- $X1 = 0.4$
- $X2 = 0.3$
- $X3 = 0.1$

Gestui atpažinti yra paduodama keletas reikšmių [X1, X2, X3].

Kad su nauja reikšme perėjo nuo A gesto būsenos į B tikimybė yra 0,3. Nuo B gesto būsenos į A tikimybė 0,4.

Turint šiuos duomenis galime sukurti šio proceso modelį:



6 pav. Paslėptojo Markovo modelio pavidys [1]

Algoritmas suskaičiuoja kiekvieno galimos būsenų sekos tikimybes.

Paskaičiuokime pavyzdį pateiktos reikšmių sekos X1, X2, X3.

- Tikimybė jog visuomet pateiktos sekos buvo būsenoje A yra $0,6 \cdot 0,1 \cdot 0,4 \cdot 0,5 = 0,012$.
- Tikimybė jog su pateiktomis sekomis būsenos kito A \rightarrow B \rightarrow A yra $0,6 \cdot 0,1 \cdot 0,3 \cdot 0,3 \cdot 0,4 \cdot 0,5 = 0,00144$.
- Tikimybė jog gestas visuomet buvo B būsenoje $0,4 \cdot 0,6 \cdot 0,3 \cdot 0,1 = 0,036$.

Būsenų sekos paskutinė reikšmė yra tos sekos rezultatas, pvz. jei būsenos buvo B,A,B jos rezultatas yra B gestas. Iš pateiktų duomenų galime spręsti jog pateiktos sekos X, X2, X3 gestas greičiausiai buvo B su tikimybe 0.036, tačiau pavyzdyje apskaičiuojame ne visus galimus variantus.

Šis algoritmas vadinamas paslėptuoju Markovo modeliu, kadangi mes nežinome koksai tai gestas, o galime tik numanyti įvertinant tikimybę. Modelyje pateiktos tikimybės gaunamos pateikus algoritmo metodams kiekvienos raidės pavyzdžius. Šis procesas vadinamas algoritmo mokymu. Mūsų kuriamos sistemos atveju, mes gauname duomenis iš „Leap Motion“ įrenginio. Šiuos duomenis mes paverčiame į atitinkamas skaičių matricas, kurios yra perduodamos algoritmui mokytis, arba atpažinti gestui. Modelio dydis yra daug kartų didesnis nei pavaizduotas brėžinyje. Visų pirmą gestų skaičius

mano kuriamoje sistemoje nėra ribojamas, o duomenų sekos gali būti skirtingų didžiųjų, bei įvairaus ilgio.

2.2. Skyriaus išvados

Nors Leap Motion įrenginys turi tam tikrų problemų kuriant rankos erdvinį modelį, jis yra plačiai naudojamas virtualios realybės grafinėms sąsajoms valdyti. Šio įrenginio programinė aplinkos pateikiamas duomenų formatas yra labai patogus gestų atpažinimui. Pateikiami duomenys lengvai gali būti transformuojami į mašinų mokymo algoritmų priimamus duomenų formatus. Mašinų mokymo algoritmams reikalingas kuo tikslesnis gestų šablonų pateikimas, todėl reikalingi duomenų filtravimo metodai. Sistemoje bus naudojami šablonų sulyginimo, bei paslėptos klasifikacijos Markovo mašinų mokymo algoritmai. Markovo algoritmas yra sukurtas atpažinti duomenų sekoms, todėl puikiai tiks rankos modelio kitimo sekai atpažinti. Nors panašūs sprendimo būdai egzistuoja, tačiau iki šiol nebuvo atlikti tyrimai naudojant Leap Motion įrenginį atpažinti lietuvių kalbos gestų abėcėlės raidžių gestus. Šie rezultatai gali padėti gerinant gestų fiksavimo, bei atpažinimo sistemų paremtų vaizdo apdorojimo tikslumą.

3. RANKOS GESTŲ ATPAŽINIMO SISTEMOS PROJEKTINĖ DALIS

3.1. Sistemos paskirtis

„Leap Motion“ įrenginys leidžia nuskaityti vartotojo rankas bei sudaryti jų trijų dimensijų modelį, tačiau ji neturi funkcionalumo, kuris galėtų atpažinti vartotojų rodomus gestus. Žinoma, yra programinės įrangos bibliotekų, kurios padeda atpažinti kai kuriuos gestus, tačiau jei norima pridėti įvairių nestandartinių gestų, kyla problema.

Šiame darbe pateikiama sistema geba išmokyti jei rodomus gestus ir vėliau juos atpažinti. Gestai sistemai yra pateikiami naudojant „Leap Motion“ įrenginį. Ši sistema remiasi mašininio mokymo algoritmu „Paslėptos klasifikacijos Markovo mokymo algoritmas“ bei šabloninio atpažinimo metodu. Sistemoje galima pasirinkti kokius metodus mokyti, kokius gestus įrašyti ar ištrinti. Sistemoje integruoti metodai registruoja visą informaciją, susijusia su gestų atpažinimu vėlesnei duomenų analizei.

Ši sistema buvo kuriama ištyrta, ar įmanoma atpažinti visas lietuvių kalbos gestų abėcėlės raides naudojant „Leap Motion“ įrenginį. Pagrindinis šios sistemos tikslas yra:

- ištyrta naudojamų algoritmų tikslumą;
- gestų mokymo sudėtingumą;
- gestų pradžios taškų radimo tikslumą;

- „Leap Motion“ įrenginio pateikiamų duomenų tikslumą.

3.2. Sukurta gestų atpažinimo sistema

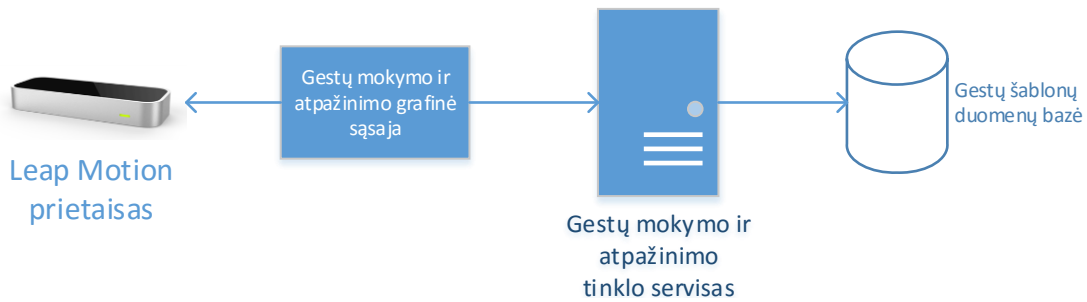
Sukurta rankos gestų atpažinimo sistema ne visada pasižymėjo dabar teikiamu funkcionalumu. Pirminė sistemos idėja buvo pokalbių sistema, kurioje susirašinėti galima būtų naudojant „Leap Motion“ įrenginį. Buvo nutarta sistemą papildyti funkcionalumu, kuris leistų raides rašyti lietuvių gestų kalbos abėcėlės raidžių gestais. Atlikinėjant gestų atpažinimo tyrimus šis modulis buvo nuolatos pildomas nauju funkcionalumu. Dėl spartaus augimo, bei didelio resursų sunaudojimu šis modulis buvo išskirtas į atskirą sistemą pavadinimu „Rankos gestų mokymo, bei atpažinimo sistema“.

Šio modulio išskyrimas į pavienę sistemą išsprendė labai daug problemų. Kadangi šis modulis tapo pakankamai didelis buvo pergaltota architektūra, suskaidant šį moduli į mažesnius komponentus. Dėl naujos architektūros sistema tapo lankstesnė naujiems pakeitimams, kurie buvo reikalingi atliekant tyrimus. Atlikus šio magistrinio darbo tyrimus sistema toliau buvo tobulinama. Ši sistema buvo nesunkiai išplėsta „Paslėptos sąlygos atsitiktinio lauko mašinų mokymo“ algoritmu. Šis algoritmas buvo mokomas tais pačiais duomenimis, kaip ir paslėpto Markovo klasifikacijos algoritmas, tačiau toks pats tikslumas nebuvo išgautas. Kadangi šiam algoritmui reikia daugiau tyrimu, bei jis nebuvo prie magistrinio darbo užsibrėžtų tikslų jis šiame darbe nepateiktas.

3.1. Gestų atpažinimo sistemos tinklo servisas

Gestus atpažinti reikalingų duomenų skaičius kyla eksponentiškai nuo gestų skaičiaus. Viršijus dešimties gestų ribą, gestų atpažinimo sistemos algoritmams apsimokyti reikia pakankamai nemažai laiko, vidutiniškai apie pusę minutės. Reikia pabrėžti jog lietuvių kalbos gestų abėcėlė susidaro iš trisdešimt dviejų gestų. Pasiėkus šią ribą gestams apsimokyti gali prireikti nuo dviejų iki keturių minučių, priklausomai nuo kompiuterio spartos. Šio darbo tyrimai reikalavo labai dažno algoritmų mokymo, dėl šios priežasties sistema buvo iškelta į tinklo servisą.

Gestų atpažinimo sistemos iškėlimas į tinklo servisą (*pav. 7*), leido nesunkiai paskirstyti sistemai reikalingus resursus tarp kelių kompiuterių. Visi gesto atpažinimo duomenys yra saugomi duomenų bazėje. Paleidus keletą servisų kurie kreipiasi į ta pačia duomenų bazę leido valdyti algoritmų mokymą. Vienu metu galima buvo įrašinėti naujus gestus, atlikinėti tyrimus, bei mokyti algoritmus atpažinti naujus gestus.



7 pav. Sistemos išdėstymo diagrama

Tyrimais buvo pagrįsta jog duomenų siuntimas iš „Leap Motion“ įrenginio per tinklą servisui labai nedidina gesto atpažinimo trukmės. Vieno gesto duomenims gauti iš „Leap Motion“ vidutiniškai susideda nuo penkių šimtų iki tūkstančio penkių šimtų skaičių. Šio kiekio duomenis perduoti tinklo servisui nereikalauja labai daug resursų ar greičio. Didžiausias sulėtėjimas įvyksta pačiame servise šiuos duomenis filtruojant, bei atliekant gesto atpažinimo funkcijas. Visas procesas neužtrunka daugiau kaip dviejų šimtų milisekundžių, dėl to vėlavimas nesijaučia. Kadangi sistemos greičio užteko tyrimui jokie papildomi metodai nebuvo taikomi spartai padidinti, tačiau keletas sprendimo būdų buvo sugalvoti jei būtų atsiradęs šis poreikis. Šie sprendimo būdai aptarti skyriuje „3.2 Galimi sistemos spartos tobulinimai“.

Tinklo servisas suteikia galimybę lengvai pasiekti gestų atpažinimo sistemą iš skirtingų aplinkų. „Leap Motion“ gali būti naudojamas žaidimuose kurie yra sukurti naudojant „Unity“ ar „Unreal Engine“ žaidimo variklius, bei nesunkiai integruotas į bet kokią Windows aplikaciją ar tinklalapį. Tačiau gestų atpažinimo algoritmai yra parašyti naudojant „C#“ programavimo kalbą, todėl į daugumą sistemų integruoti juos būtų pakankamai sudėtinga. Gestų atpažinimo sistema paleista kaip tinklo servisas leidžia labai daug skirtingų sistemų komunikuoti su ja. Šiuo metu dėl lengvesnio įdiegimo buvo naudojamas SOAP protokolas, tačiau sistemą lengvai galima praplėsti jog ji priimtų ir REST duomenų formato užklausas. Šis funkcionalumas leistų gestų atpažinimo sistemą pasiekti iš bet kokios aplinkos.

3.2. Galimi sistemos spartos tobulinimai

Iš „Leap Motion“ gauti duomenys yra saugomi objektų sąrašuose. Šie sąrašai yra paverčiami į atitinkamą formatą kai yra siunčiami per tinklą servisui. Gavus šiuos sąrašus servisas juos paverčia vėl į objektų sąrašus. Jei kliento pusėje šie objektų sąrašai būtų paversti į dvimatį masyvą, šie duomenys būtų kur kas greičiau siunčiami per tinklą. Servise esantys filtravimo algoritmai šiuos duomenis taip pat greičiau apdorotų, jei jiems būtų pritaikomi matricų algoritmai. Tačiau visas kodas taptų kur kas sudėtingesnis ir sunkiau skaitomas. Filtravimo algoritmai taptų sunkiai modifikuojami. Naudojant objektus galima tiksliai pasakyti ką kokia reikšmė reiškia pagal jos pavadinimą. Matricos

pavidale sunku pasakyti kokią reikšmę atitinkamoje matricos vietoje skaičius turi. Klaidos atveju būtų sunku pasakyti kurie duomenys sukėlė problemą. Kadangi naudojat šią sistemą reikės iširti gestų atpažinimo tikslumą naudojant mašininio mokymo algoritmus yra kur kas svarbiau lengvai modifikuojamas ir skaitomas kodas, nei greitesnė sistema. Rezultatų failus taip pat lengviau generuoti naudojant objektų sąrašus. Ne gana to ši sistema buvo pakankamai sparti ir jos tobulinimas greičio atvejų būtų aktualus jei ji turėtų aptarnauti labai daug klientų.

3.3. Gestų šablonų saugojimas, bei filtravimas

Su „Leap Motion“ įrenginiu įrašyti duomenys yra saugomi „MSSQL“ duomenų bazėje. Kuriant bei testuojant sistemą buvo pastebėta, jog mašinių mokymo algoritmai yra ypač jautrūs įvedamiems duomenims. Dėl šios priežasties gestai buvo suskirstyti į kategorijas. Lengviausia yra atpažinti ar gestas yra atliekamas su judesiu ar be. Jei pateikti duomenys išlieka tie patys, vadinasi gestas yra be judesio. Gestai su judesiu ir be jo yra saugojami skirtingose duomenų bazės lentelėse. Taip išskirstyti duomenys leidžia sukurti du to paties algoritmo objektus. Vienas būtų mokamas gestais su judesiu, o kitas be judesio. Šis duomenų suskirstymas labai pagerino sistemos gestų atpažinimo kokybę, bei davė naudingų įžvalgų gerinti sistemą.

Suskirstyti duomenys į kategorijas leido taikyti skirtingus filtravimo metodus gestų su judesio ar be atpažinimui. Gestams kurie neturi judesio atpažinti yra nesvarbu kokioje erdvės vietoje yra atliekamas gestas. Vadinasi iš pateiktų duomenų mes galime išimti X,Y,Z ašių koordinatčių duomenis. Atpažinti duomenims užtenka posvyrio, pokrypio ir tangažo. Šiems gestams taip pat nereikia labai daug rankos pavienių kadru, kadangi gestas išlieka rimtyje. Vadinasi jei gestas yra nejudantis, mes galime paimti tik pirmąjį kadrą. Šiomis dvejomis sąlygomis yra išfiltruojama labai daug nereikalingų duomenų, kurios keldavo problemų mašinių mokymo algoritmui.

Judantiems gestams nebuvo atliekama labai daug filtravimo algoritmų, kadangi beveik visi duomenys yra labai svarbūs. Ašių X,Y,Z koordinatės buvo pastumtos link nulio, kadangi yra svarbu jog gesto seka būtų pradedama iš tos pačios padėties. Markovo paslėptos klasifikacijos algoritmas puikiai tinka atpažinti duomenų sekoms, dėl to daugiau klasifikacijos bei filtravimo metodų šiam algoritmui pateikiamiems duomenims nereikėjo.

3.4. Mašinių mokymų algoritmų pritaikymas

Priklausomai ar gestas yra su judesiu ar be yra taikomi skirtingi atpažinimo metodai. Kuriant sistemą, bei atliekant tyrimus buvo pastebėta, jog resursų ir atpažinimo kokybės atžvilgiu gestus kurie yra atliekami su judesiu geriausiai yra atpažinti naudojant paprasto sulyginimo su šablonu metodą. Jei bent viena reikšmė yra nukrypusi nuo leistinos normos gestas nėra atpažįstamas. Atpažinti nejudančius

gestus neprireikė sudėtingesnių mašinių mokymų algoritmų kadangi jų atpažinimui nereikalinga sudėtinga logika.

Gestai su judesiu susideda iš daugiau nei vieno kadro. Kuriant sistemą buvo bandoma šiuos gestus atpažinti naudojant paprastą šablonų sulyginimą. Šis būdas buvo labai neefektyvus, bei sudėtingas kadangi sunku buvo nustatyti kokį kadrą su koku šabloniniu kadru lyginti. Taip pat kadru kiekiai nuolatos skyrėsi, kadangi gestas galėjo būti atliekamas greičiau ar lėčiau. Šiai problemai spręsti buvo pasitelktas Markovo paslėptos klasifikacijos algoritmas, kuris skirtas atpažinti duomenų sekoms. Su šiuo algoritmu buvo rasta labai gerų pavyzdžių, kurie sprendė panašias problemas, pvz. raidžių atpažinimas pagal skirtingą rašysena. Šio algoritmo mokymas buvo pakankamai sudėtingas, kadangi jis reikalavo tikslių duomenų. Taip pat šis algoritmas pasižymėjo permokymo problema. Jei algoritmui atpažinti vieną gestą yra skiriama labai daug duomenų, algoritmas pradeda atpažinti tik tą gestą.

Dėl šių priežasčių prie sistemos buvo pridėtas papildomos funkcijos kurios padėjo suvaldyti algoritmo mokymą. Visų pirma pridėta funkcija kuri leido pasirinkti, kokius gestus pateikti mokymo algoritmas išmokti. Ši funkcija buvo naudinga, kai buvo norima pridėti keletą naujų gestų, ir nenorima jog sistema mokytus atpažinti visus, kadangi tai užtrukdavo labai daug laiko. Palaipsniui mokant sistemą atpažinti tik keletą gestų galima greitai pastebėti kada buvo įvesti netinkami duomenys, bei kada duomenų pakanka gestui atpažinti. Taip pat buvo pridėta testavimo funkciją. Pridedant naujus gestus nežinoma ar sistema neregresavo. Norint tai patikrinti galima įrašyti į duomenų bazę testavimo gestus. Kai yra išmokoma atpažinti naują gestą, galima sistemai paduoti testavimo gestus, bei laukiamus rezultatus. Jei šie gestai vis dar yra atpažįstami tinkamai, naujas gesto pridėjimas nesugadino atpažinimo algoritmo.

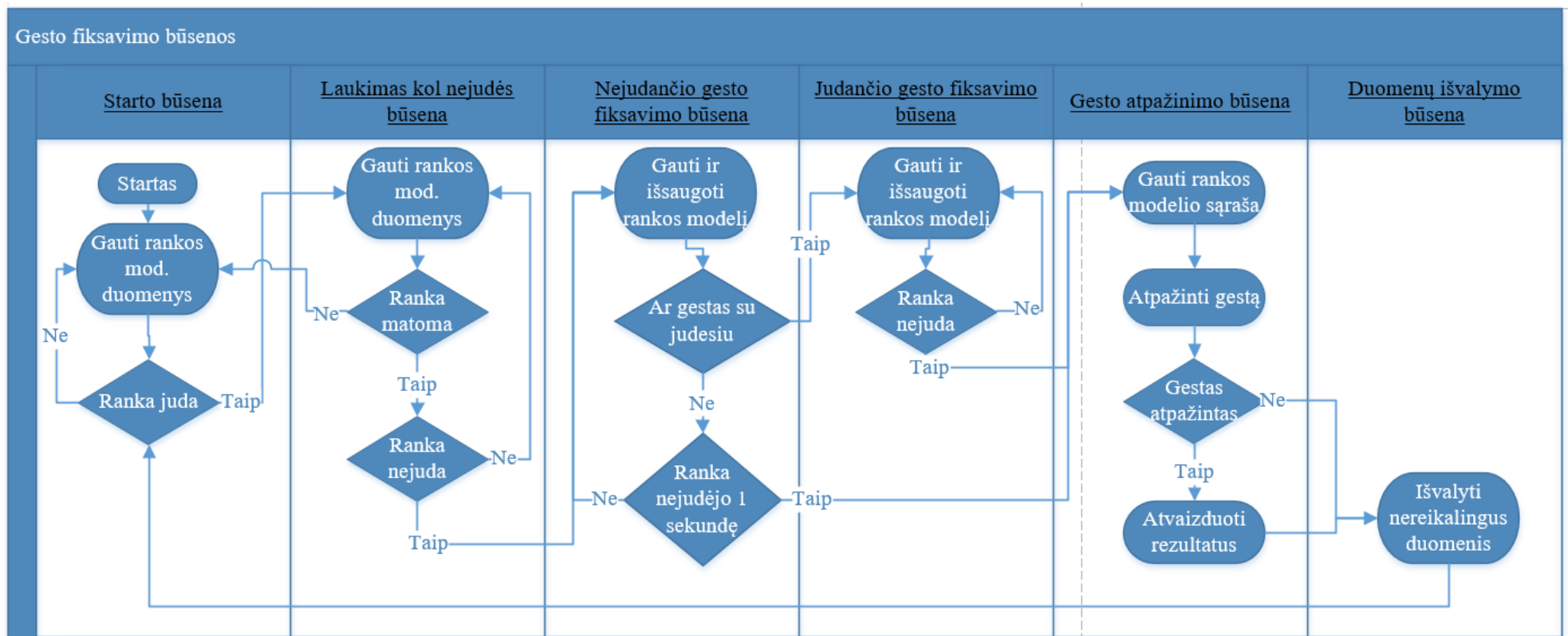
3.5. Gestų fiksavimas

Anksčiau minėjome jog sistemos korektiškam veikimui reikalingi tikslūs duomenys iš „Leap Motion“ įrenginio. „Leap Motion“ nuolatos pateikia vartotojo rodomos rankos kadrus. Problema kyla kai norime iš šių kadru eilės išfiltruoti kada yra pradedamas rodyti gestas. Jei gestų atpažinimo sistemai siųstume visus duomenis, sistema labai dažnai atpažins šiuos gestus. Šis reiškinys pasireiškia todėl, kad tam tikri gestai gali susidaryti iš kelių kito tipo gestų. Tai dažnai pastebima rodant gestą su judesiu. Gestas su judesiu yra sudarytas iš daug kadru, tarp kurių pasitaiko gestų be judesio.

Norint išspręsti šia problemą parašytas metodas kuris fiksuoja vartotojo rankos judesius. Sistema atitinkamu metu gali būti vienoje iš šešių būsenų (*pav. 8*).

- Starto būseną. Sistema laukia kol vartotojo ranka pradės judėti. Jeigu ranka pradėjo judėti pereinama į laukimo kol nejudės būseną.

- Laukimas kol nejudės būseną. Jei sistema nemato rankos, būseną gražinama į starto. Jei užfiksuojama jog vartotojas nejudina rankos pereinama į nejudančio gesto fiksavimo būseną.
- Nejudančio gesto fiksavimo būseną. Vartotojas nejudina rankos dvi sekundes, yra užfiksuojamas nejudantis gestas. Įrašyti rankos modelio duomenys yra išsaugomi ir pereinama į gesto atpažinimo būseną. Jeigu vartotojas per dvi sekundes pajudina ranka sistemos būseną pasidaro judančio gesto fiksavimo.
- Judančio gesto fiksavimo būseną. Jei yra užfiksuojama, kad vartotojas jau nebejudinta rankos, užfiksuoti rankos modelio duomenys yra išsaugoti, bei sistemos būseną pakeičiama į gesto atpažinimo.
- Gesto atpažinimo būseną. Šioje būsenoje užfiksuoti duomenys yra išsiunčiami gestų atpažinimo posistemei. Kai posistemei gražina rezultatus, jie yra pateikiami vartotojui, bei pereinama į duomenų išvalymo būseną.
- Duomenų išvalymo būseną. Išvalomi nereikalingi duomenys, bei pereinama į „Starto“ būseną.



8 pav. Gestų fiksavimo posistemės būsenų diagrama

3.6. Funkciniai reikalavimai

Svarbiausios sistemos funkcijos:

- Įrašyti gestus į duomenų bazę. Vartotojas privalo turėti galimybę įrašyti rankos gestą į duomenų bazę. Gestai skirtingiems atpažinimo algoritmams yra saugomi skirtingose duomenų bazėse. Vartotojas gali pasirinkti į kokio algoritmo duomenų bazę įrašyti gestą.
- Ištrinti gestus iš duomenų bazės. Vartotojas gali ištrinti gestus iš pasirinktos duomenų bazės.
- Mokyti algoritmus. Vartotojas privalo turėti galimybę sistemai pasakyti kada ir kokiems algoritmams reikia apsimokyti iš esamų duomenų.
- Gestu atpažinimo metodai. Gestai privalo būti atpažįstami pagal du metodus: šablonų sulyginimo, bei paslėptos klasifikacijos Markovo mokymo algoritmo.
- Sistemos funkcijų pakeitimas. Sistema gali įrašinėti gestus į duomenų bazę arba juos atpažinti. Vartotojas gali pasirinkti ką šiuo metu daro sistema.
- Gesto patvirtinimas. Įrašius gestą sistema pateikia koksai tai gestas. Jei norima galima šio gesto neįrašyti į duomenų bazę ir pašalinti iš sąrašo.
- Gestų atpažinimas. Jei sistema yra nustatyta atpažinti gestus, vartotojo rodomi gestai yra fiksuojami realiu laiku. Išvedamoje informacijoje parodoma koksai algoritmas kokį gestą mato.

3.7. Nefunkciniai reikalavimai

3.7.1. Reikalavimai sistemos išvaizdai

Bendri reikalavimai gestų įvedimo, bei atpažinimo sistemos sąsajai:

- Lengvai suprantama sąsaja
- Informuojanti vartotoją apie atliktus veiksmus sąsaja
- Prisitaikanti prie skirtingų rezoliucijų vaizduoklių sąsaja
- Profesionalios išvaizdos sąsaja
- Sąsaja neerzina vartotojo (pvz. nereikalingais patvirtinamais atliktų veiksmų)

3.7.2. Reikalavimai sistemos išvaizdai

Panaudojimo paprastumas gestų įvedimo, bei atpažinimo sistemai:

- Lengvai pastebimos klaidos
- Paprasta naudotis įvairų išsilavinimą turintiems asmenims (paprasti ne mokslinio tipo žymėjimai sąsajoje).
- Galimybė suderinti su kitomis sistemomis

3.7.3. Reikalavimai vykdymo charakteristikoms

Užduočių vykdymo greičio apribojimai:

- Algoritmams mokyti reikalingas laikas negali būti ribojamas
- Sistema negali priimti gesto atpažinimo užklausų kol algoritmai mokosi.

- Maksimalus gesto užfiksavimo laikas iki 10 sekundžių.
- Maksimalus gesto atpažinimo laikas iki 1 sekundės.
- Maksimalus gesto šablono įvedimo laikas iki 1 sekundės.

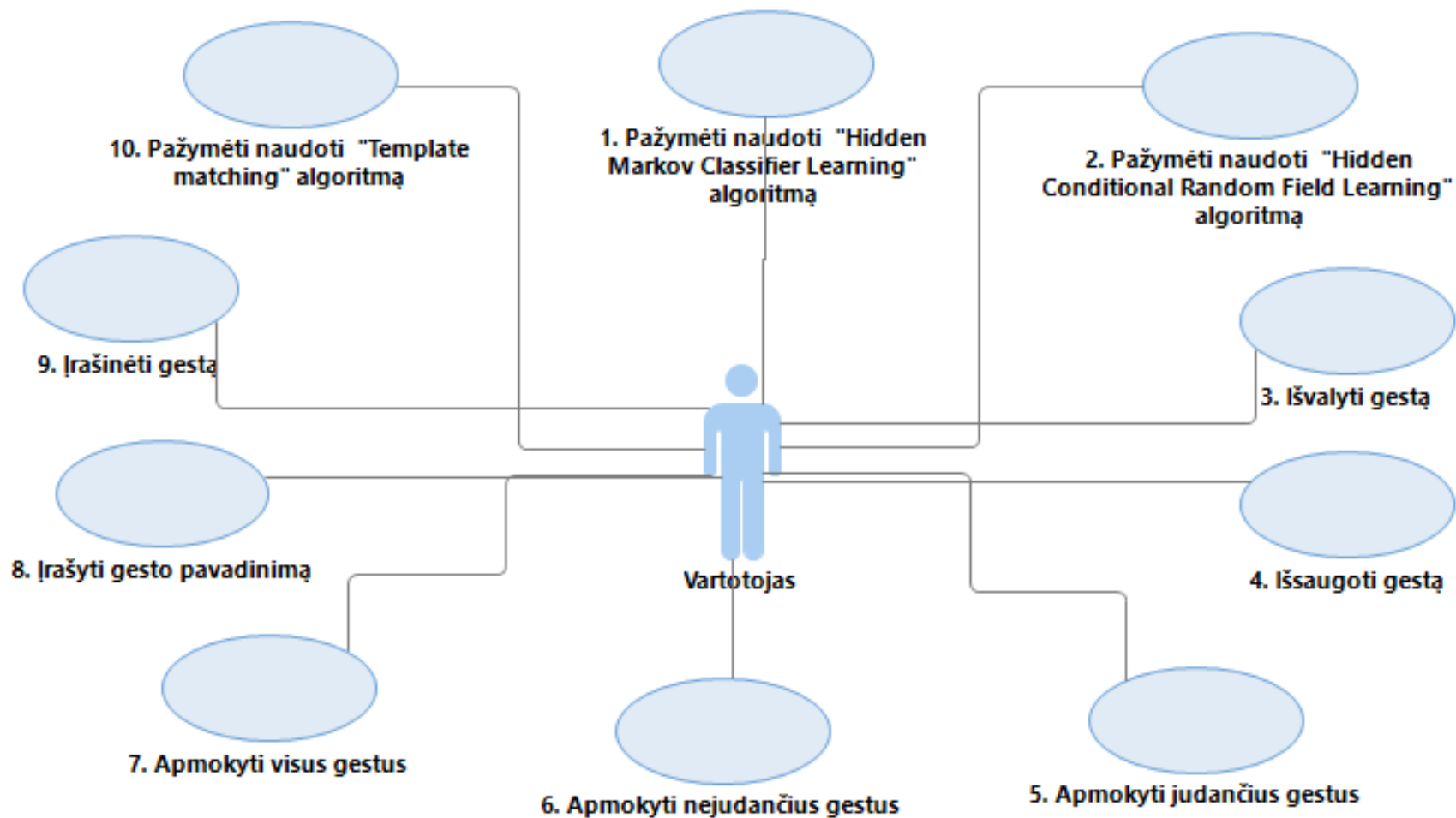
3.7.4.Reikalavimai veikimo sąlygoms

- Sistema turi veikti Windows 32 ar 64 bitų operacinėje sistemoje, kurios versija gali būti: „Windows 7“, „Windows 8“, „Windows 10“.
- Sistemai veikti reikalingas prie kompiuterio prijungtas „Leap Motion“ įrenginys. Taip pat kompiuteryje turi būti įdiegta „Leap Motion“ įrenginio programinė įranga.
- Sistemai reikalinga „MSSQL“ duomenų bazė gestų šablonams saugoti.

3.7.5.Reikalavimai saugumui

- Vientisumas. Sistemoje nebus saugoma jokia privati vartotojo informacija. Sistemoje bus saugoti tik vartotojo rankos erdvinio modelio duomenys tik tokiu atveju jei vartotojas juos įrašys į duomenų bazę. Vartotojas galės ištrinti rankos erdvinį modelį iš duomenų bazės jei to pageidaus. Sistema gali būti prijungta prie interneto dėl gestų šablonų išsaugojimo į duomenų bazę.
- Pasiekiamumas. Pagrindiniai sistemos duomenys su kuriais dirba sistema yra rankos erdvinio modelio šablonai kurie yra saugomi duomenų bazėje. Šie duomenys negali būti pasiekiami trečioms šalims be sistemos kūrėjo sutikimo. Gestų atpažinimo, bei įvedimo sistema veiks kaip internetinis servisas todėl gali būti pasiekiamas kitų sistemų. Sistemos funkcionalumo pasiekiamumas turi būti apsaugotas.
- Konfidencialumas. Sistema nuskenuos tik vartotojo ranką, norint sudaryti rankos erdvinį modelį. Jokių papildomų duomenų sistema nesaugos.

3.8. Sistemos panaudos atvejų diagrama



9 pav. Sistemos panaudos atvejų diagrama

Panaudos atvejis Nr. 1	Pažymėti naudoti „Hidden Markov Classifier Learning“ algoritmą
Tikslas	Pažymėti naudoti šį algoritmą, kai reikės mokyti sistemą
Aktoriai	Vartotojas
Ryšys su kitais PA	5,6,7
Nefunkciniai reikalavimai	Nėra
Prieš-sąlyga	Nėra
Sužadinimo sąlyga	Vartotojas nori išsaugoti gestą į "Hidden Markov Classifier Learning" algoritmo duomenų bazę.
Po-sąlyga	Vartotojas pažymi, jog norės išsaugoti gestą į "Hidden Markov Classifier Learning" algoritmo duomenų bazę.
Pagrindinis scenarijus	Programa išsimena, jog vartotojas norės išsaugoti gestus į "Hidden Markov Classifier Learning" algoritmo duomenų bazę.
Alternatyvus scenarijus	Nėra

Panaudos atvejis Nr. 2	Pažymėti naudoti "Hidden Conditional Random Field Learning" algoritmą
Tikslas	Pažymėti naudoti šį algoritmą, kai reikės mokyti sistemą
Aktoriai	Vartotojas
Ryšys su kitais PA	5,6,7
Nefunkciniai reikalavimai	Nėra
Prieš-sąlyga	Nėra
Sužadinimo sąlyga	Vartotojas nori išsaugoti gestą į "Hidden Conditional Random Field Learning" algoritmo duomenų bazę.
Po-sąlyga	Vartotojas pažymi, jog norės išsaugoti gestą į "Hidden Conditional Random Field Learning" algoritmo duomenų bazę.
Pagrindinis scenarijus	Programa išsimena, jog vartotojas norės išsaugoti gestus į "Hidden Conditional Random Field Learning" algoritmo duomenų bazę.
Alternatyvus scenarijus	Nėra

Panaudos atvejis Nr. 3	Išvalyti gestą
Tikslas	Ištrinti atitinkamą gestą iš duomenų bazės
Aktoriai	Vartotojas
Ryšys su kitais PA	Nėra
Nefunkciniai reikalavimai	Nėra
Prieš-sąlyga	Nėra
Sužadinimo sąlyga	Vartotojas nori ištrinti gestą iš pasirinktų duomenų bazių.
Po-sąlyga	Vartotojas ištrina gestą iš pasirinktų duomenų bazių.
Pagrindinis scenarijus	<ol style="list-style-type: none"> 1) Programa gauna gesto pavadinimą iš įvedimo laukelio. 2) Programa išsiunčia tinklo servisui, kokį nori gestą ištrinti bei iš kokių duomenų bazių. 3) Tinklo servisas ištrina gestą iš pasirinktų duomenų bazių.
Alternatyvus scenarijus	Nėra

Panaudos atvejis Nr. 4	Išsaugoti gestą
Tikslas	Išsaugoti gestą į duomenų bazę
Aktoriai	Vartotojas
Ryšys su kitais PA	Nėra
Nefunkciniai reikalavimai	Nėra
Prieš-sąlyga	Nėra
Sužadinimo sąlyga	Vartotojas nori išsaugoti gestą į pasirinktas duomenų bases.
Po-sąlyga	Vartotojas išsaugo gestą į duomenų bases
Pagrindinis scenarijus	<ol style="list-style-type: none"> 1) Programa gauna duomenis iš „Leap Motion“. 2) Programa gauna gesto pavadinimą 3) Programa konvertuoja duomenis į reikalingą formatą 4) Programa išsiunčia duomenis į tinklo servisą. 5) Servisas išsaugo gesto duomenis į atitinkamas duomenų bases
Alternatyvus scenarijus	Nėra

Panaudos atvejis Nr. 5	Mokyti judančius gestus
Tikslas	Mokyti judančių gestų atpažinimo algoritmus.
Aktoriai	Vartotojas
Ryšys su kitais PA	Nėra
Nefunkciniai reikalavimai	Nėra
Prieš-sąlyga	Nėra
Sužadinimo sąlyga	Vartotojas nori praleisti visus judančių gestų mokymo algoritmus.
Po-sąlyga	Vartotojas praleidžia visus judančių gestų mokymo algoritmus. Judančių gestų algoritmai atpažins gestus pagal naujus duomenis iš duomenų bazės.
Pagrindinis scenarijus	1) Programa kreipiasi į tinklo servisą, jog jis praleistų visus judančių gestų mokymo algoritmus 2) Algoritmai atnaujinami.
Alternatyvus scenarijus	Nėra

Panaudos atvejis Nr. 6	Mokyti nejudančius gestus
Tikslas	Mokyti nejudančių gestų atpažinimo algoritmą
Aktoriai	Vartotojas
Ryšys su kitais PA	Nėra
Nefunkciniai reikalavimai	Nėra
Prieš-sąlyga	Nėra
Sužadinimo sąlyga	Vartotojas nori praleisti visus nejudančių gestų mokymo algoritmus.
Po-sąlyga	Vartotojas praleidžia visus nejudančių gestų išmokymo algoritmus. Nejudančių gestų algoritmai atpažins gestus pagal naujus duomenis iš duomenų bazės.
Pagrindinis scenarijus	1) Programa kreipiasi į tinklo servisą, jog jis praleistų visus nejudančių gestų mokymo algoritmus 2) Algoritmai atnaujinami.
Alternatyvus scenarijus	Nėra

Panaudos atvejis Nr. 7	Mokyti visus gestus
Tikslas	Mokyti visus atpažinimo algoritmus
Aktoriai	Vartotojas
Ryšys su kitais PA	Nėra
Nefunkciniai reikalavimai	Nėra
Prieš-sąlyga	Nėra
Sužadinimo sąlyga	Vartotojas nori praleisti visus mokymo algoritmus.
Po-sąlyga	Vartotojas praleidžia visus mokymo algoritmus. Algoritmai atpažins gestus pagal naujus duomenis iš duomenų bazės.
Pagrindinis scenarijus	1) Programa kreipiasi į tinklo servisą jog jis praleistu visus mokymo algoritmus 2) Algoritmai atnaujinami.
Alternatyvus scenarijus	Nėra

Panaudos atvejis Nr. 8	Įrašyti gesto pavadinimą
Tikslas	Įrašyti gesto pavadinimą
Aktoriai	Vartotojas
Ryšys su kitais PA	Nėra
Nefunkciniai reikalavimai	Nėra
Prieš-sąlyga	Nėra
Sužadinimo sąlyga	Vartotojas įrašyti gesto pavadinimą
Po-sąlyga	Vartotojas įrašo gesto pavadinimą
Pagrindinis scenarijus	Programa išsimena gesto pavadinimą
Alternatyvus scenarijus	Nėra

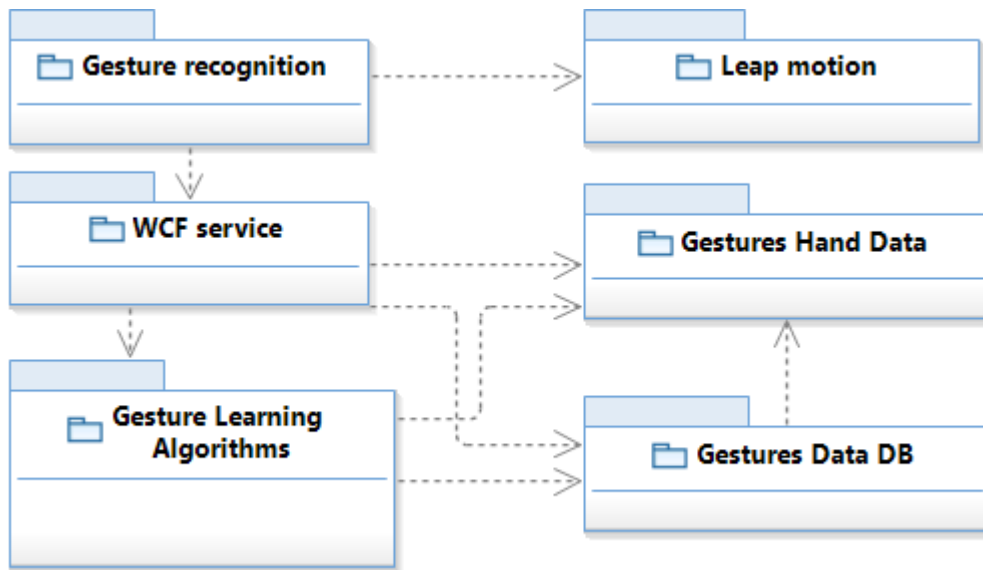
Panaudos atvejis Nr. 9	Įrašinėti gestą
Tikslas	Įrašinėti gestą
Aktoriai	Vartotojas
Ryšys su kitais PA	P5
Nefunkciniai reikalavimai	Nėra
Prieš-sąlyga	Nėra
Sužadinimo sąlyga	Vartotojas nori įrašinėti duomenis gautus iš „Leap Motion“.
Po-sąlyga	Vartotojas gauną iš Leap motin duomenų masyvą, kuris reprezentuoja atitinkamą gestą.
Pagrindinis scenarijus	Gaunami duomenis iš „Leap Motion“ yra įrašinėjami į masyvą, kol įrašinėjimo mygtukas yra paspaustas.
Alternatyvus scenarijus	

Panaudos atvejis Nr. 10	Pažymėti naudoti "Template matching" algoritimą
Tikslas	Pažymėti naudoti šį algoritimą, kai reikės mokyti sistemą.
Aktoriai	Vartotojas
Ryšys su kitais PA	Nėra
Nefunkciniai reikalavimai	Nėra
Prieš-sąlyga	Nėra
Sužadinimo sąlyga	Vartotojas nori išsaugoti gestą į „Template matching“ algoritmo duomenų bazę.
Po-sąlyga	Vartotojas pažymi, jog norės išsaugoti gestą į „Template matching“ algoritmo duomenų bazę.
Pagrindinis scenarijus	Programa išsimena jog vartotojas norės išsaugoti gestus į „Template matchign“ algoritmo duomenų bazę.
Alternatyvus scenarijus	Nėra

3.9. Sistemos architektūros modelis

3.9.1. Apžvalga

Paveikslėlyje žemiau pateikta rankos gestų atpažinimo sistemos, suskirstytos į paketus, vaizdas.



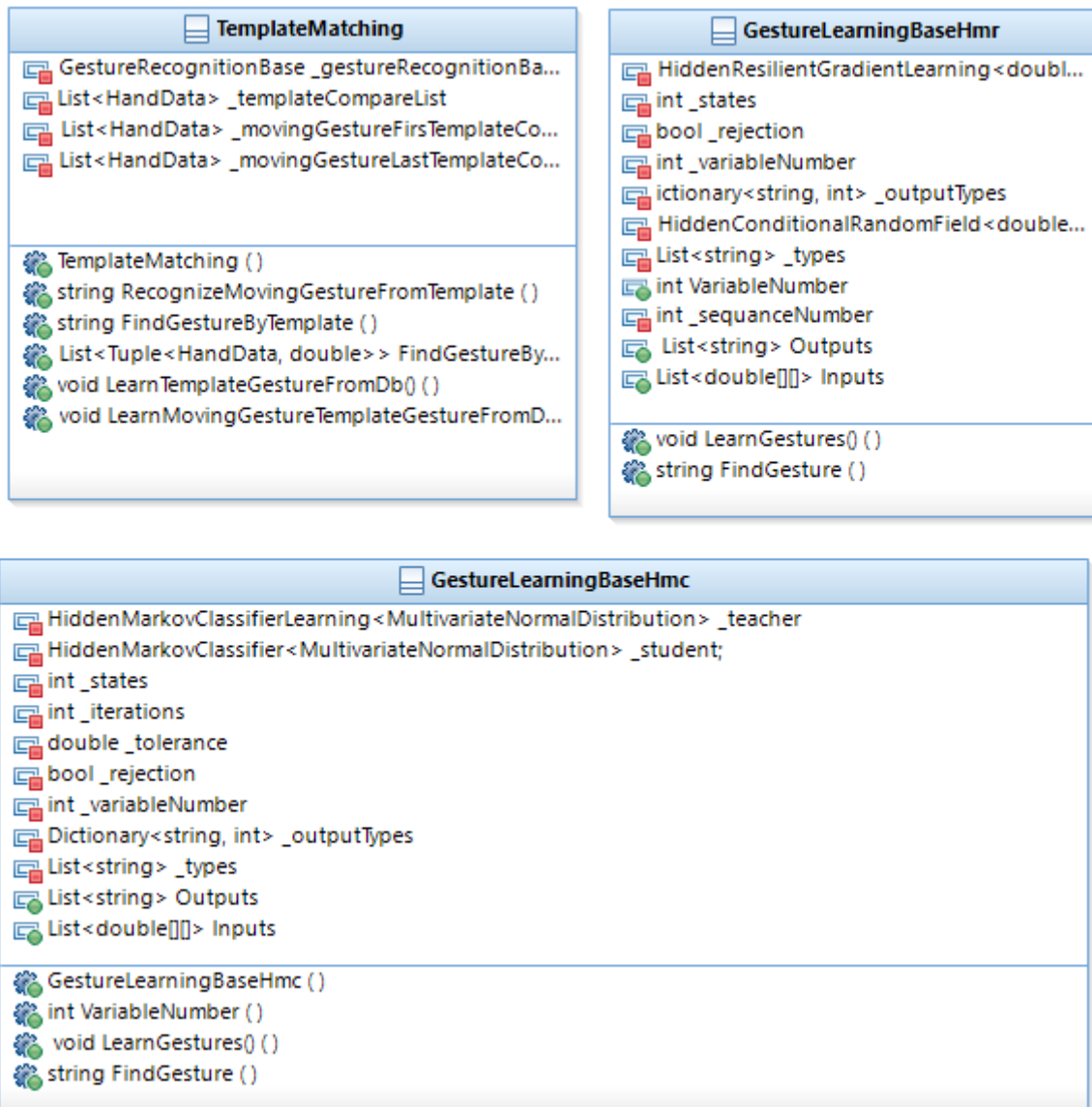
10 pav. Sistemos paketų diagrama

3.9.2. Paketų detalizavimas

Paketas „Gesture learning algorithms“

Šiame pakete galima rasti gestu atpažinimo metodų klases:

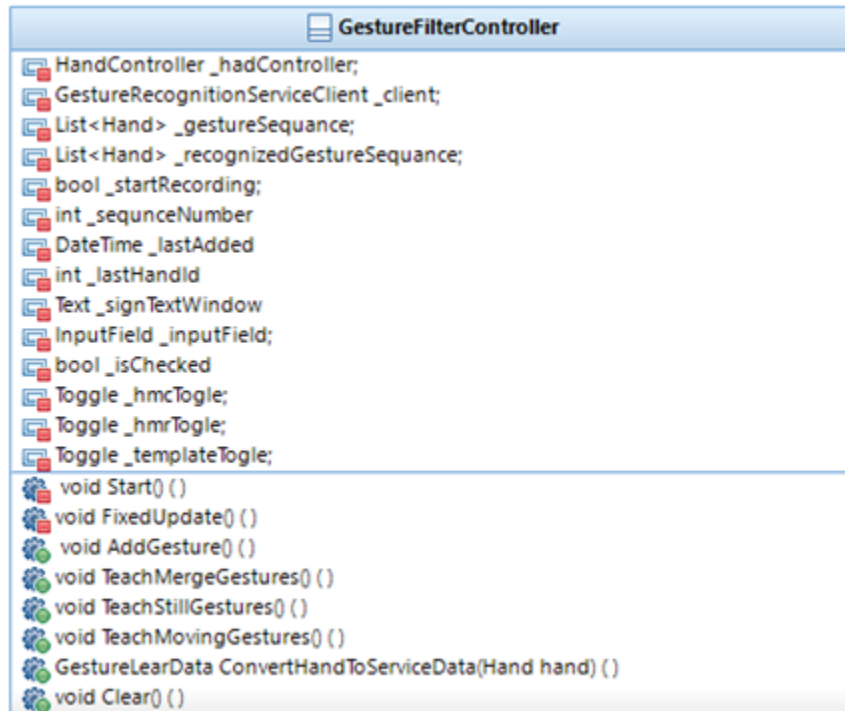
- „Hidden Markov Classifier Learning“ algoritmo klasė;
- „Hidden Condition Random Field Learning“ algoritmo klasė;
- „Template matching“ algoritmo klasė.



11 pav. Paketo „Gesture Learning Algorithms“ klasių diagrama.

Paketas „Gesture recognition“

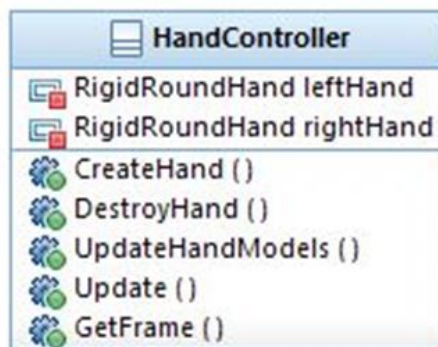
Pagrindinės sistemos klasių diagrama. Joje duomenis paimami iš „Leap Motion“ prietaiso, bei siunčiami tinklo servisui apdorojimui. Šioje klasėje yra saugoma informacija, kokie algoritmai turi būti atnaujinti, bei kokius gestus ištrinti iš sistemos. Taip pat šiame pakete esančios klasės atsakingos už gautų duomenų atvaizdavimą.



12 pav. Paketo "Gesture Recognition" klasių diagrama.

Paketas „Hand controller“

Šis paketas yra sukurtas „Leap Motion“ kompanijos. Šio paketo pagalba iš „Leap Motion“ gaunami reikalingi duomenys. Šis paketas turi daugiau klasių, tačiau duomenims gauti naudojame tik šią klasę.

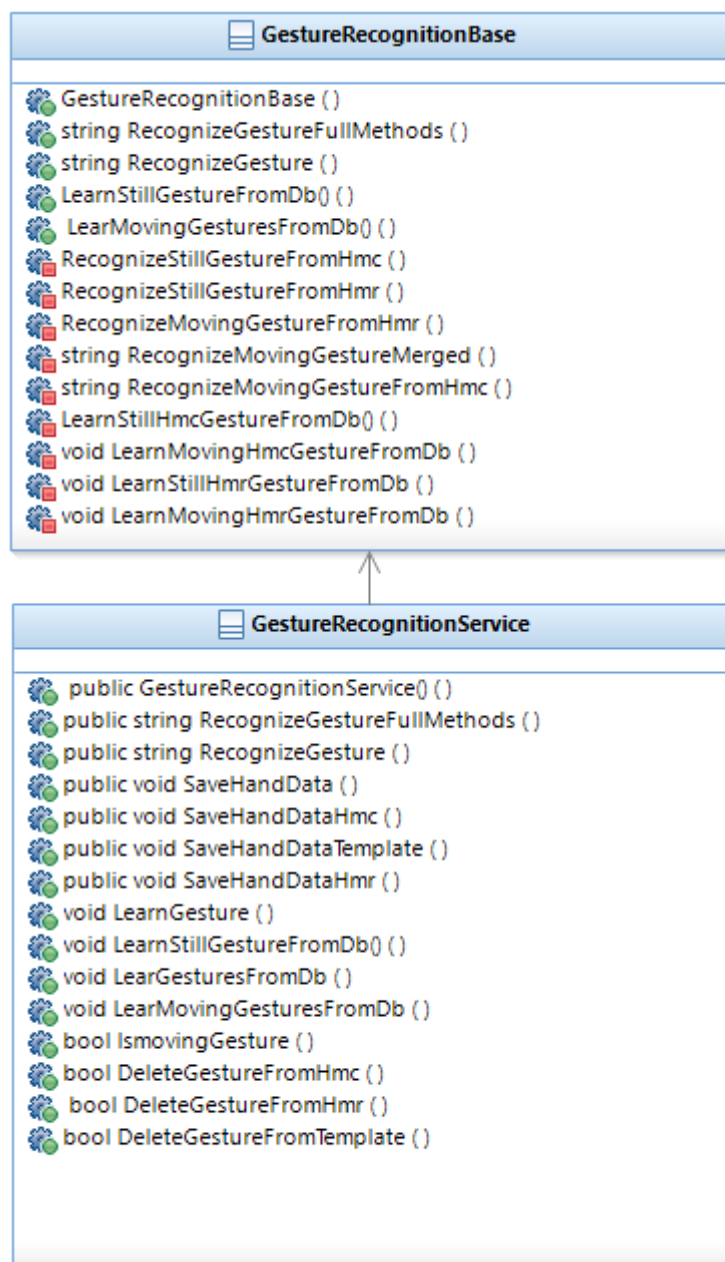


13 pav. Paketo "„Leap Motion“" klasių diagrama.

Paketas „WCF Service“

Tinklo serviso pagrindinis paketas. Jame galima rasti dvi klases. Viena iš jų atsakinga už komunikaciją su pagrindine sistema. Pagrindinė sistema kreipiasi į šios klasės metodus, kai reikia atnaujinti atpažinimo algoritmus, ištrinti gestus, ar juos atpažinti.

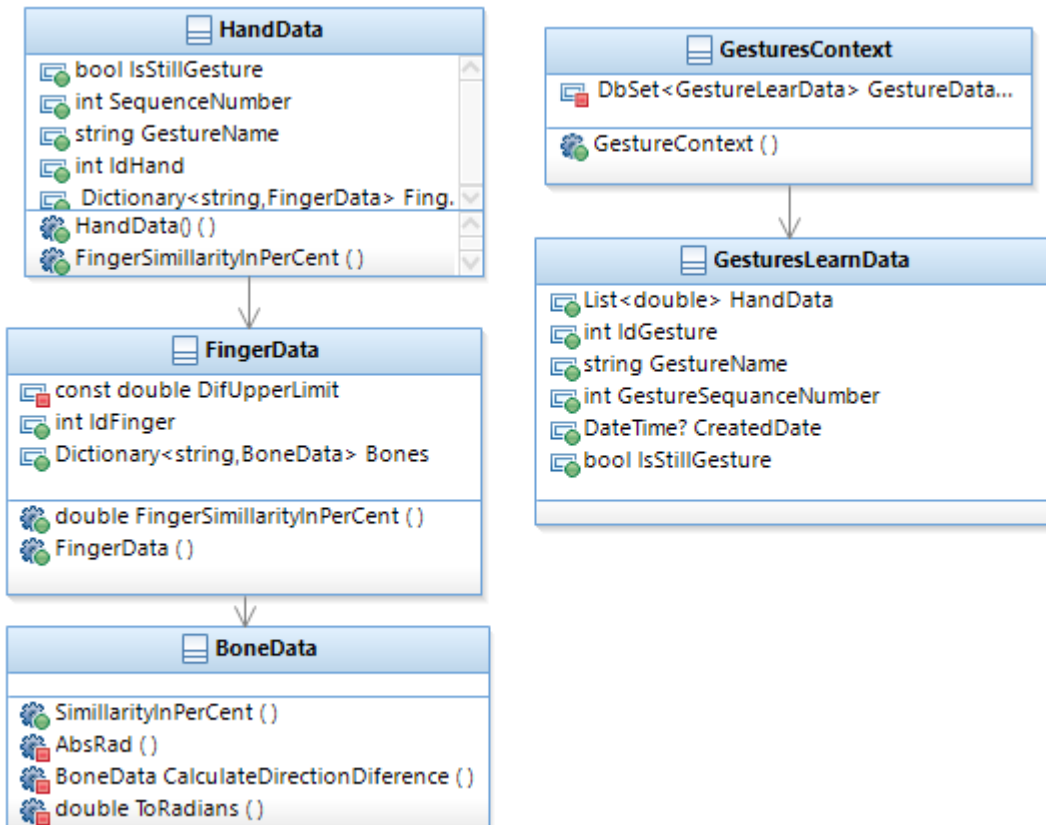
Antroji klasė tai pagrindinė gestų atpažinimo klasė, kurioje pagal gautus duomenis sprendžiama kokius algoritmus naudoti, kokia informaciją išsaugoti, bei kokią ištrinti. Šioje klasėje saugoma gestų atpažinimo logika.



14 pav. Paketo "WCF Service" klasių diagrama

Paketas „Gesture hand data“

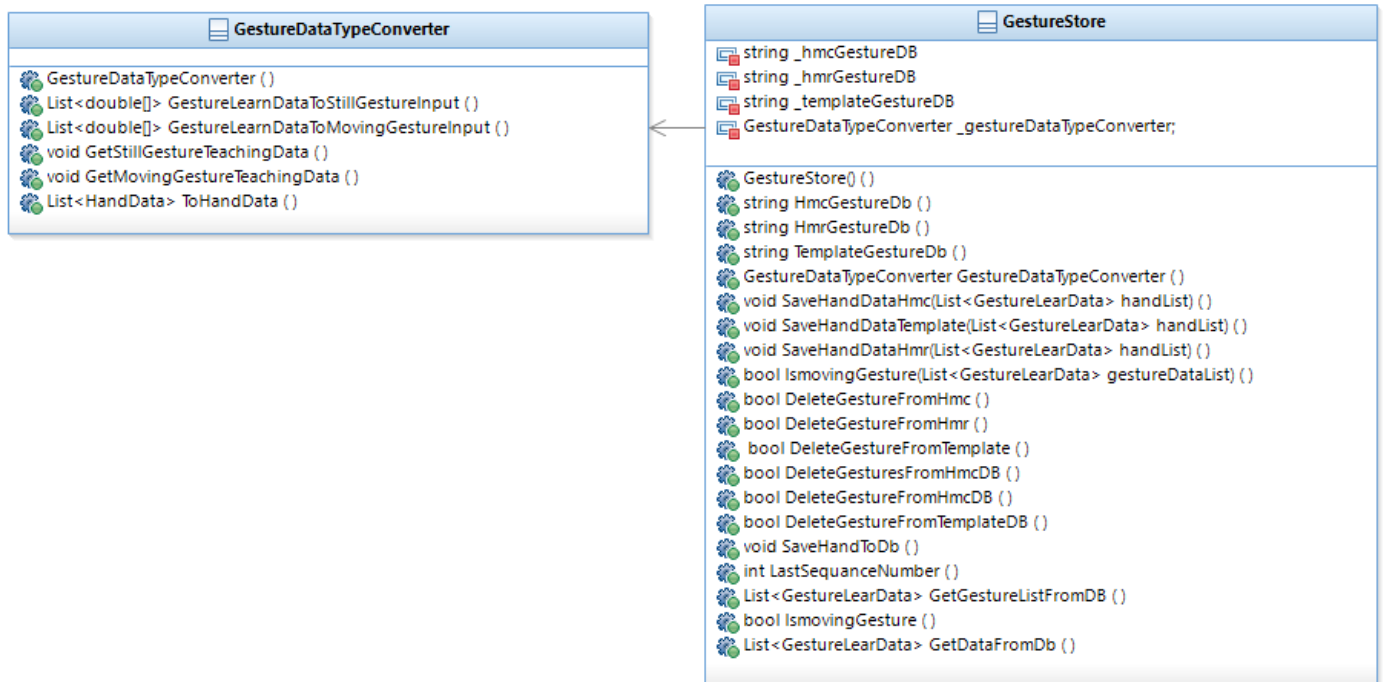
- Šiame pakete saugomos konteinerio tipo klasės, duomenis iš „Leap Motion“ kaupiti, bei apdoroti.
- Konteinerinės klasės: HandData, bei GesturesLearnData.
- GestureContext tai ryšio su duomenų bazės lentelėmis klasė.



15 pav. Paketo "Gestures Hand Data" klasių diagrama.

Paketas „Gestures data db“

- Šiame pakete yra klasės atsakingos už duomenų išsaugojimą į duomenų bazę, bei duomenų transformavimą iš vieno tipo į kitą.
- GestureDataTypeConverter klasėje yra metodai, kurie atsakingi už duomenų konvertavimą iš vieno tipo į kitą.
- GestureStore klasė atsakinga už duomenų išsaugojimą duomenų bazėje, jų gavimą ar ištrynimą.



16 pav. Paketo "Gestures Data DB" klasių diagrama.

3.10. Naudojami trečiųjų šalių komponentai

„Accord Framework“ [27] – tai „C#“ programavimo kalba parašyta mašinių mokymo algoritmų programinės įrangos biblioteka. Šiame pakete galima rasti mašinių mokymų, bei kitų matematinių algoritmų realizacijų. Ši biblioteka buvo naudojama dėl Markovo paslėptos klasifikacijos mašinių mokymo algoritmo realizacijos.

„Unity“ [28] - žaidimų kūrimo variklio programinė įranga. Ji buvo naudojama vizualiai parodyti rankos erdvinį modelį, bei su ja kurta grafinė vartotojo sąsaja.

„Unity Leap Motion Assets“ [29]– įskiepis į „Unity“ sistemą gauti iš „Leap Motion“ rankos erdvinį modelį, bei jį parodyti vizualiai „Unity“ sistemoje.

3.11. Naudojami aparatūrinė įranga

„Leap Motion“ (17 pav.)- įvesties įrenginys paremtas vaizdo atpažinimo technologijomis. Šis įrenginys fiksuoja žmogaus rangas, bei sukuria jų erdvinį modelį. Šis įrenginys turi dvi monochromines infraraudonųjų spindulių (IS) kameras ir tris IS šviesos diodus. Iš dviejų 2D vaizdų, gautų monochrominėmis IS kameromis, „Leap Motion“ sugeneruoja rankų erdvinį modelį. Šio modelio pagalba galima valdyti įvairias programines sistemas, panašiai kaip su kompiuterine pele. Priešingai nei Microsoft kompanijos kuriamas Kinect, kuris traktuoja pilną žmogaus skeletą, „Leap Motion“ seka tik vartotojų rankas. Įvertinant žmogaus rankos galimybes, „Leap Motion“ programinė įranga gali numanyti pirštų, delno ar riešo padėtis ir tuo atveju, kai kameros šių vietų nemato. Rankas „Leap Motion“ gali traktuoti nuo 25 iki 600 milimetrų atstumu, 150 laipsnių kampu, leidžiant vartotojui laisvai jaustis erdvėje. Nors ir šios charakteristikos yra svarbios, pats didžiausias „Leap Motion“ privalumas yra jo programavimo sąsaja, kuris leidžia lengvai kurti taikomąją programinę įrangą. Šis įrenginys yra dažnai naudojamas valdyti įvairioms virtualios realybės grafinėms sąsajoms.



17 pav. Leap Motion įrenginys

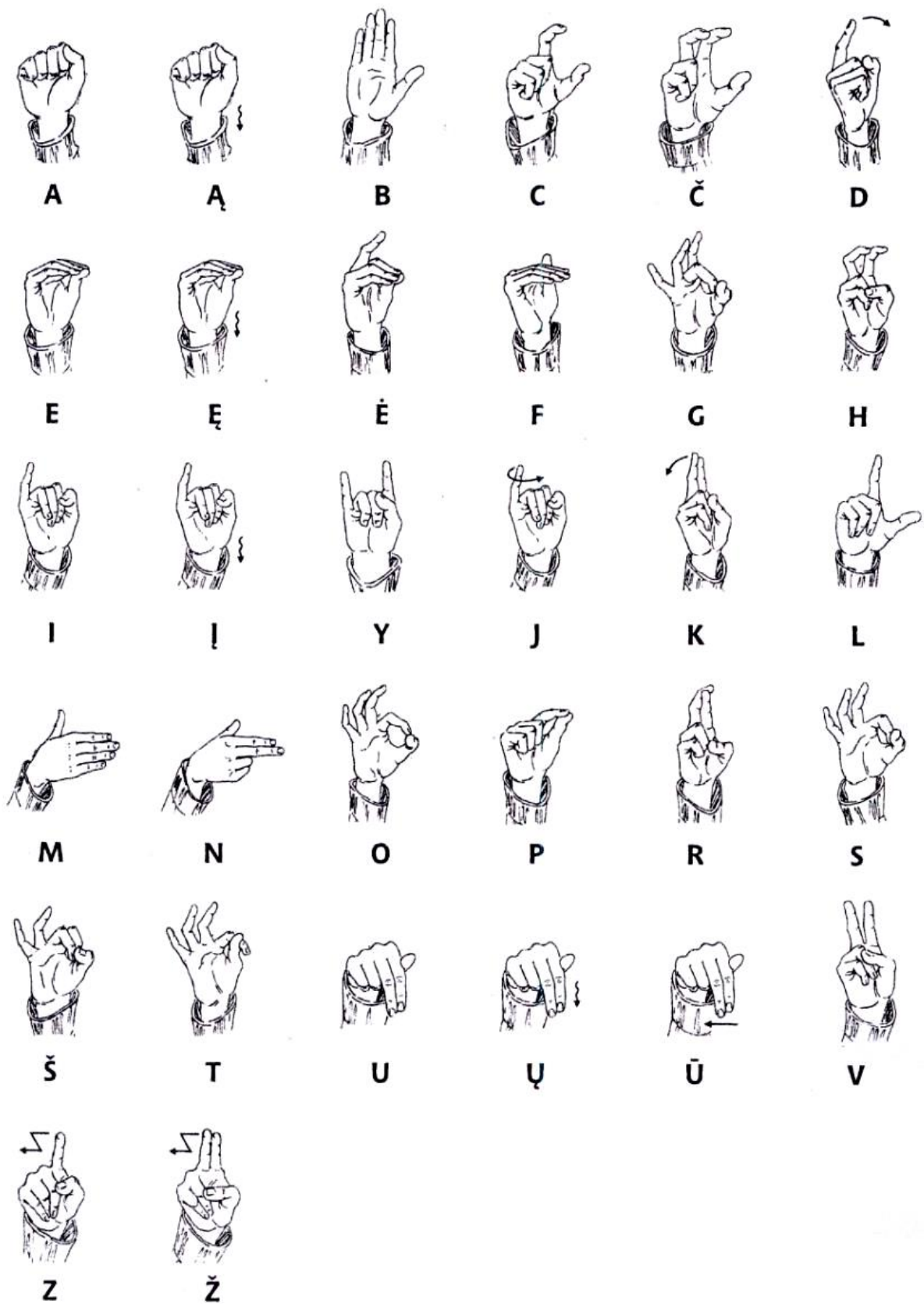
4. RANKOS GESTŲ ATPAŽINIMO SISTEMOS TYRIMAI

Sukurta rankų gestų atpažinimo sistema gauna duomenis iš Leap Motion įrenginio, kurio veikimas paremtas vaizdo atpažinimu. Vaizdo atpažinimo įrenginiai pasižymi aklos zonos problema, t. y. gali matyti tik dalį sekamo objekto. Leap Motion įrenginio atveju rodant gestą dažnai nematoma dalis rankos. Šias nematomas rankos vienas Leap Motion programinė įranga įvairiais matematiniais algoritmais bando apskaičiuoti. Sistemos tyrimo tikslas yra patikrinti sistemos tikslumą naudojantis šiuo įrenginiu, bei surasti kurių gestų atlikimas kelia daugiausiai problemų. Sistema gali atpažinti gestus ir tuo atveju jei rodomas gestas yra netikslus, tačiau šiuo atveju algoritmams išmokyti turi būti pateikti ir atitinkamų netikslių gestų šablonai.

4.1. „Leap Motion“ sugeneruoto rankos modelio atvaizdavimo tikslumas

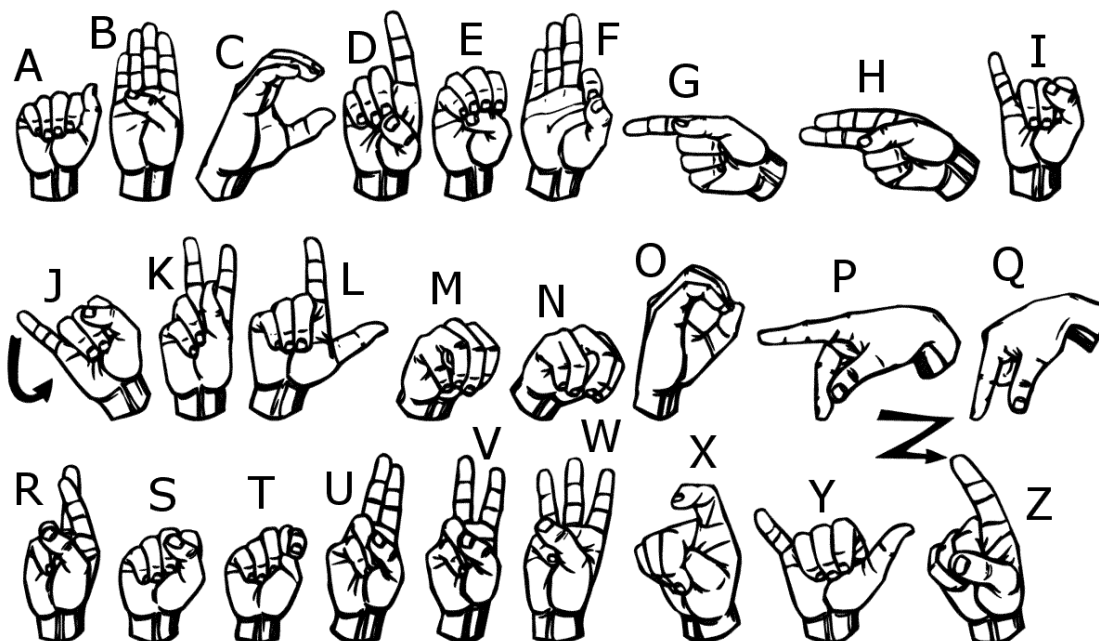
Gestų atpažinimo sistemos tikslumas labai priklauso nuo jei pateikiamų duomenų. Prieš kuriant gestų atpažinimo sistemą buvo išanalizuotas „Leap Motion“ įrenginio rankos modelio atvaizdavimo tikslumas. Šis tyrimas suteikė labai daug svarbių duomenų apie „Leap Motion“ veikimo principą, kokias problemas kelia, bei kaip jas galima spręsti. Tyrime buvo naudojami lietuvių, bei amerikiečių kalbos gestų abėcėlė. Tyrimas atliktas su kolega Mantu Taroza, Kauno Technologijos Universiteto, „Programų sistemos inžinerijos“ magistro studijų studentu.

Lietuvių gestų kalbos abėcėlė (*18 pav.*) susideda iš trisdešimt dviejų gestų. Jei norima pabrėžti, jog raidė yra su nosinę, tos raidės gestas yra atliekamas su bangos judesiu. Ši abėcėlė turi devynis gestus kurie yra atliekami su tam tikru judesiu. Ši abėcėlė buvo naudojama ir tolimesniems tyrimams su sukurta gestų kalbos atpažinimo sistema.



18 pav. Lietuvių kalbos gestų abėcėlės raidžių gestai

Amerikiečių gestu kalbos abėcėlė susideda iš dvidešimt šešių gestų. Ši abėcėlė turi tik du gestus kurie yra atliekami su judesiu, tačiau joje yra daugiau vienas su kitu panašių gestų.



19 pav. Amerikiečių gestu kalbos abėcėlės raidžių gestai

4.1.1. Amerikiečių gestu kalbos abėcėlės atpažinimas

Eksperto dalyviai turėjo atlikti amerikietiškos gestu kalbos abėcėlės raidžių gestus (19 pav). Kiekvienos raidės gestas buvo atliktas dešimt kartų. Atliktus dalyvių rankų gestus mes užregistravome „Leap Motion“ aplinkoje, bei nufotografavome realiai rankų rodytus gestus. Vėliau šio tyrimo duomenys buvo analizuojami norint patikrinti ar „Leap Motion“ teisingai užfiksavo atliktus gestus.

Išanalizavus duomenis, pateiktus lentelėje (Lentelė Nr. 1), pastebėta, kad yra problemų aptinkant tarpą tarp pirštų. Vieno centimetro tarpai yra blogai traktuojami ir sunku nustatyti kada rodomas C, o kada O gestas.

Gestai, kuriems reikalinga tiksli nykščio padėtis, taip pat yra sunkiai nustatomi. Nykštys yra dažnai uždengiamas kitų pirštų. Nykščio padėtį „Leap Motion“ bando nuspėti. Gaunama apytikslė padėtis, kurios neužtenka tiksliam raidžių E, M, N, T, H, K, S, V, X gestų traktavimui.

Pagrindinės problemos, pastebėtos tyrimo metu, yra tos, jog kai kuriems gestams atpažinti reikia labai tikslaus 3D rankos erdvinio vaizdo. Tai akivaizdžiai pasireiškė atliekant P raidės gestus, kai pirštai yra tik dalinai užlenkti (nesugniaužti į kumštį), tačiau iš delno pusės įrenginys nustatė pirštus kaip pilnai užlenktus. Taip pat ši problema pasireiškė su raidės R gestu, t.y. pirštai turi būti sukryžiuoti, bet jie yra pateikiami kaip suglausti, o toks gestas atitinka U raidę. „Leap Motion“ nustato žmogaus

rankos erdvinę formą atsižvelgiant į pirštus. Problema pasireiškė, kai rodomas sugniaužtos rankos gestas ir „Leap Motion“ nesugebėdavo tiksliai nustatyti rankos pasukimo kampo ir dėl to atvaizduodavo ją pasuktą neteisingu kampu. Toks įrenginio elgesys taip pat būdingas ir jau aptartais atvejais, kai dalinai užlenkti pirštai traktuojami kaip pilnai užlenkti. Tokiu atveju taip pat ranka yra pasukama, nors toks veiksmas nėra atliekamas.

Lentelė nr. 1. Leap Motion amerikiečių gestų kalbos raidžių atpažinimo duomenys

Raidė	Atpažinimas,%	Raidė	Atpažinimas,%	Raidė	Atpažinimas,%
A	94	J	70	S	46
B	48	K	44	T	8
C	50	L	80	U	62
D	66	M	10	V	66
E	54	N	0	W	86
F	76	O	70	X	56
G	82	P	56	Y	86
H	66	Q	32	Z	88
I	70	R	46		

4.1.2. Lietuvių gestų kalbos abėcėlės atpažinimas

Šios tyrimo dalies metu eksperimento dalyviai turėjo atlikti lietuvių gestų kalbos abėcėlės raidžių gestus (*18 pav*). Kiekvienos raidės gestas buvo atliktas dešimt bandymų. Kaip ir ankstesnės tyrimo dalies metu, atlikti gestai buvo užregistruoti ir lyginami su realiai rodomu rankos gestu. Dauguma lietuvių kalbos abėcėlės gestų yra panašūs į amerikiečių kalbos gestus. Rezultatai pateikti lentelėje (Lentelė Nr. 2).

Lentelė nr. 2. „Leap Motion“ lietuvių gestų kalbos raidžių atpažinimo duomenys

Raidė	Atpažinimas,%	Raidė	Atpažinimas,%	Raidė	Atpažinimas,%	Raidė	Atpažinimas,%
A	100	Ė	70	K	44	Š	96
Ą	100	F	70	L	80	T	70
B	88	G	88	M	100	U	50
C	30	H	56	N	50	Ū	52
Č	54	I	70	O	100	Ū	70
D	72	Į	90	P	45	V	64
E	88	Y	86	R	46	Z	88
Ę	94	J	92	S	60	Ž	95

Atlikus tyrimą, pastebėtos tos pačios problemos atpažįstant gestus, kurios pasireiškė su amerikiečių gestų kalbos abėcėle. Problemos atsiranda tuomet, kai „Leap Motion“ įrenginys nemato tam tikrų pirštų. Tuomet jų padėčių negalima užfiksuoti ir jie yra identifikuojami netinkamai. Jei pirštai yra prastai matomi, jie yra blogai pateikiami erdvėje.

Lietuvių kalbos abėcėlė yra sudėtingesnė nei amerikiečių, kadangi turi daugiau gestų, kurie reikalauja papildomo judesio, pavyzdžiui Ȧ ar Ū̇ raidžių gestai. Tokius gestus yra sudėtingiau aptikti, kai jie yra greitai rodomi vienas po kito.

4.2. Sukurtos sistemos tyrimai, bei tolimesni tikslai

Sukurtos sistemos tikslumas buvo tiriamas ja išmokant atpažinti visus lietuvių gestų kalbos abėcėlės raidžių gestus. Pradžioje buvo įvedama kiekvienai raidei po tris gesto šablonus. Po šablonų įvedimo buvo tikrinama kaip atpažįstamas kiekvienas gestas. Jei gestas buvo prastai atpažįstamas, šiam gestui buvo įvedama daugiau gestų šablonų. Šis procesas atliktas apie dešimt kartų, kol sistemos tikslumas tapo pakankamai stabilus, jog būtų galima naudotis sistema. Reikia paminėti, kad gestai kai kuriais atvejais būdavo visiškai ištrinami iš sistemos duomenų bazės, bei pateikiami nauji jų šablonai. Į šablonų duomenų bazėje įvesta penkiasdešimt vienas gestų su judesiu šablonų, bei septyniasdešimt vienas gestų be judesio šablonas. Viso duomenų bazėje saugomi keturi šimtai aštuoniasdešimt devyni rankos erdvinio modelio kadrai.

Lentelė nr. 3. Gestų su judesiu šablonų skaičius

Gestas	šablonų skaičius
Ȧ	5
D	3
Ė	3
İ	4
J	3
K	10
U̇	4
Ū̇	8
Z	6
Ž	5

Lentelė nr. 4. Gestų be judesio šablonų skaičius

Gestas	šablonų skaičius
A	3
B	5
C	5
Č	4
E	3
É	3
F	3
G	3
H	3
I	3
Y	3
L	3
M	3
N	3
O	2
P	3
R	3
S	3
Š	3
T	3
U	4
V	3

Su dauguma raidžių gestų tikslumas viršija devyniasdešimt penkis procentus. Dėl „Leap Motion“ problemų raidžių Ū, Ū, Č, M, N, F, E, E gestų erdviniai modeliai būna nekorektiški, dėl to sistemos atpažinimo tikslumas nukenčia. Raidžių I, bei J gestai pradžioje buvo labai tarpusavyje maišomi atpažinimo algoritmo. Bandymų būtu buvo rasti tinkami šių gestų šablonai, jog algoritmas juos tiksliai atskirtų.

Atrinkti tinkamus gestų šablonus užtrunka labai daug laiko. Mokyti sistema aptikti šiuos lietuvių kalbos gestų abėcėlės raidžių gestus užtruko apie keturias valandas. Sistema buvo išbandyta tik sistemos kūrėjo, dėl to reikalingi papildomi tyrimai su kitais vartotojais. Tyrimo metu labai svarbu fiksuoti:

- gestų atpažinimo tikslumą;
- „Leap Motion“ įrenginio keliamas problemas;
- šablonų skaičių kitimą;
- vartotojo streso lygi atlikinėjant gestus;
- teksto įvedimo naudojant lietuvių kalbos gestų abėcėlės raidžių gestus greitį, bei sudėtingumą.

Turint šia informaciją sistema galima būtų pridėti automatinius šablonų filtravimo metodus. Sistema visus šablonus saugotų pagrindinėje duomenų bazėje. Įvykdžius šiuos šablonų filtravimo metodus galima būtų tinkamus šablonus įrašyti į algoritmų mokymo duomenų bazę. Šio funkcionalumo privalumas yra tas, jog pridėjus naujų gestų į sistemą, ji gebėtų pastebėti jog šie nauji gestų šablonai gali kelti bėdų atpažįstant naujus gestus. Sistema gebėtų pati išanalizuoti jei pateiktą informaciją, kadangi rankinis kiekvieno gesto šablonų analizavimas yra labai varginantis ir sunkus procesas.

4.3. Tyrimui reikalingas sistemos funkcionalumas

Prie sistemos buvo pridėtos papildomos rankos gestų atpažinimo tyrimui reikalingos funkcijos. Sistema papildyta įvestų gestų atpažinimo testavimo funkcija. Vartotojas galės tam tikrus atliktus gestus pridėti prie gestų testavimo šablonų duomenų bazės. Testavimo metu gestų šablonai bus paduodami atpažinimo algoritmams. Gavus iš algoritmų rezultatus bus sulyginama ar algoritmas šį gestą atpažino tinkamai. Šie testai reikalingi patikrinti ar įvedus naujus gestus algoritmas neregresuoja ir vis dar sugeba tinkamai atpažinti anksčiau įvestus gestus. Šis procesas turėtų pagreitinti tinkamų šablonų atrinkimo procesą, bei suteiks naudingos informacijos tobulinant sistemą.

Prie sistemos pridėti metodai kurie įrašinės į failus visą su gestų duomenimis susijusia informaciją. Metodai įrašinės:

- kokie gestai yra įvedami;
- automatizuotų gestų atpažinimo testų rezultatai;
- kiek kartų atitinkamas gestas buvo ištrintas;

- kiek laiko užtruko mokytis algoritmus, bei su kokiais duomenimis buvo apmokoma.
- kita sekančiame skyriuje pateikiama informacija

Visa informacija buvo įrašinėjama į atitinkamus „CSV“ formato failus. Šis formatas buvo pasirinktas, kadangi jį palaiko „Excel“ programinė įranga, su kuria šiuos rezultatus galima buvo lengvai peržiūrėti, bei apipavidalinti.

5. RANKOS GESTŲ ATPAŽINIMO SISTEMOS EKSPERIMENTINĖ DALIS

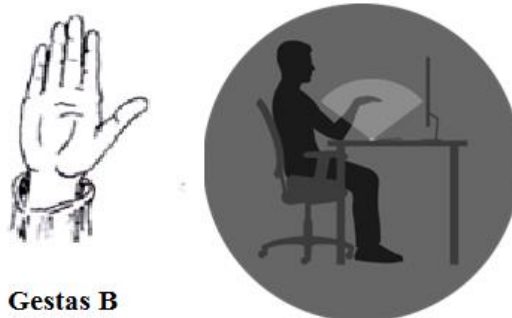
5.1. Mašinių algoritmų mokymas atpažinti lietuvių gestų kalbos abėcėlę raidžių gestus

5.1.1. Tyrimo tikslas

Pirmojo tyrimo tikslas yra ištirti kiek lietuviu kalbos abėcėlės raidžių gestams reikia pateikti šablonų, jog sistema galėtų atpažinti kiekviena gestą, bei kaip šį šablonų skaičių galima būtų sumažinti. Šio tyrimo taip pat norima pastebėti, kaip keičiasi jau įvestų gestų šablonų skaičius pridėdant naujus gestus. Tyrimo metu labai svarbu stengtis sistemai pateikti kuo tikslesnius šablonus, norint sužinoti su koku minimaliu šablonų skaičiumi sistema geba efektyviai dirbti.

5.1.2. Pasiruošimas tyrimui

Sukurtos sistemos gestų šablonų duomenų bazė buvo išvaloma. Patikrinama ar sistema negražina jokių duomenų rodant gestą, kadangi ji neturi turėti jokių gestų šablonų pavyzdžių. Ant stalo vartotojui pastatomas „Leap Motion“ įrenginys. Vartotojas turės rodyti gestus, jog rankos plaštaką būtų į „Leap Motion“ prietaisą. Pavyzdys kaip vartotojas atlieka gestą B pavaizduotas 20 paveikslėlyje. Prieš eksperimentą vartotojai apsiprato naudotis „Leap Motion“ įrenginiu, parodant visų lietuvių gestų kalbos abėcėlės raidžių gestus.



Gestas B

5.1.3. Tyrimo eiga.

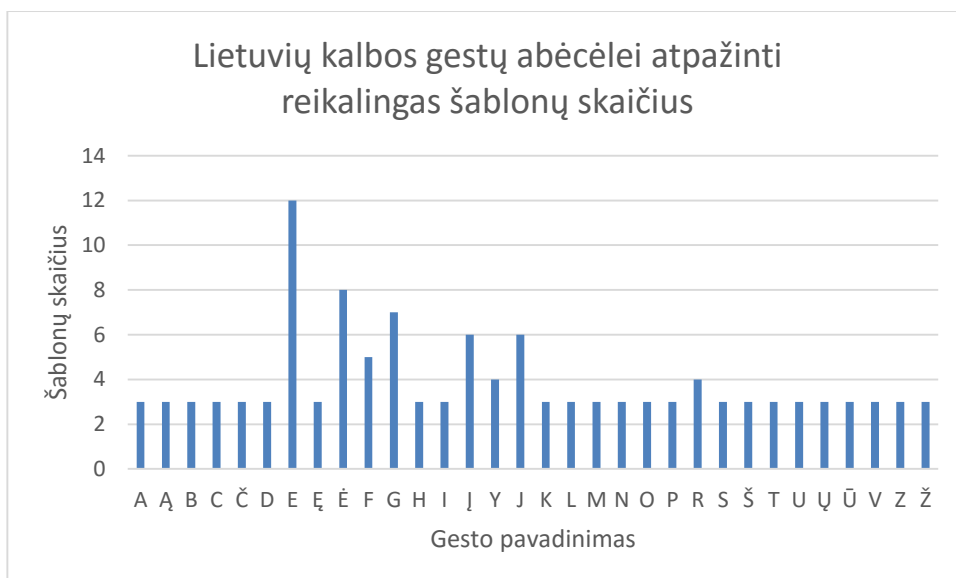
Eksperimentas buvo atliekamas su visomis lietuvių kalbos abėcėlės raidėmis. Algoritmai palaipsniui buvo mokomi atpažinti abėcėlės gestus. Vienu metu algoritmai iškart buvo išmokomi vidutiniškai keturių gestų. Kiekvienam gestui pateikiama po tris šablonus, kaip tas gestas turi būti atliekamas. Išmokius algoritmą naujų gestų buvo tikrinama ar gestas juos tiksliai atpažįsta. Jei netiksliai, algoritmui buvo pateikiama daugiau šablonų, kaip tas gestas turi būti atliekamas. Kai algoritmas naujus gestus sugebėdavai tiksliai atpažinti, kiekvienam gestui buvo sukuriama testai. Šie testai reikalingi patikrinti, ar įvedus kitus gestus algoritmas neregresuoja, ir vis dar sugeba atpažinti anksčiau įvestus gestus. Jei algoritmas regresuodavo, buvo atliekami keli veiksmai. Visų pirmą buvo bandoma gestams kurie yra atpažįstami netinkamai įvesti daugiau šablonų. Jei tas nepadėdavo tekdavo ištrinti tą gestą ir algoritmą išmokyti iš naujo jį atpažinti arba pašalinti naujojo gesto šablonus kurie užgoždavo senąjį gestą. Visi duomenys buvo automatiškai fiksuojami sistemos ir įrašomi į failą. Šie duomenys vėliau buvo išanalizuoti, bei pateikti lentelės forma.

5.1.4. Tyrimo rezultatai

Lentelėje nr. 5 pavaizduoti pirmojo tyrimo rezultatai. Stulpeliuose pavaizduota kiek reikėjo gestų šablonų algoritmui mokyti juos atpažinti. Kai kuriems gestams atpažinti šablonų kiekis viršijo numatytąjį minimalų. Šių gestų eilutės yra pavaizduotos pilkai.

Lentelė nr. 5. Pirmojo tyrimo rezultatai. Gestui atpažinti reikalingas šablonų skaičius.

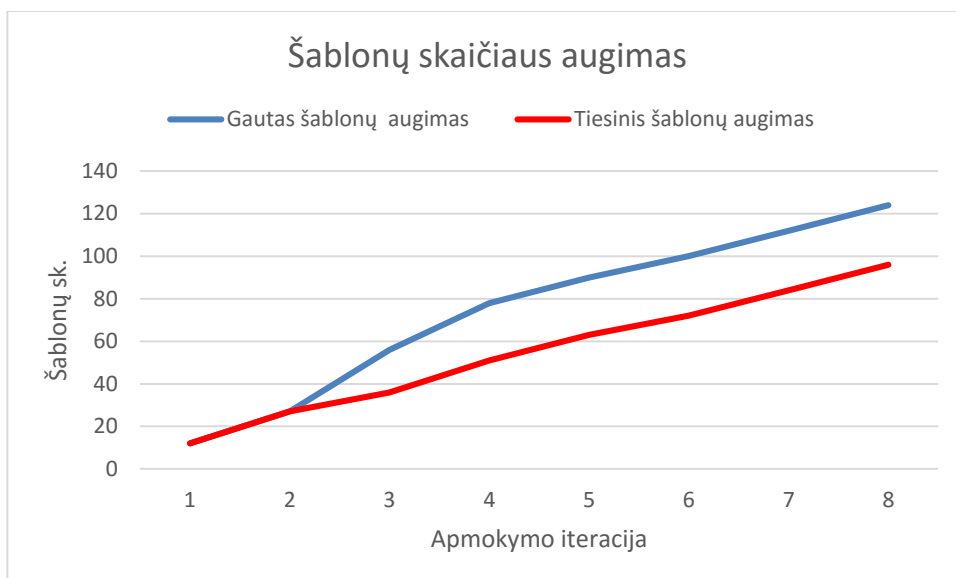
Gestas	Apmok.1	Apmok.2	Apmok. 3	Apmok. 4	Apmok. 5	Apmok. 6	Apmok. 7	Apmok 8
A	3	3	3	3	3	3	3	3
Ą	3	3	3	3	3	3	3	3
B	3	3	3	3	3	3	3	3
C	3	3	3	3	3	3	3	3
Č		3	3	3	3	3	3	3
D		3	3	3	3	3	3	3
E		3	12	12	12	12	12	12
Ę		3	3	3	3	3	3	3
Ė		3	8	8	8	8	8	8
F			5	5	5	5	5	5
G			7	7	7	7	7	7
H			3	3	3	3	3	3
I				3	3	3	3	3
J				6	6	6	6	6
Y				4	4	4	4	4
J				6	6	6	6	6
K				3	3	3	3	3
L					3	3	3	3
M					3	3	3	3
N					3	3	3	3
O					3	3	3	3
P						3	3	3
R						4	4	4
S						3	3	3
Š							3	3
T							3	3
U							3	3
U							3	3
Ū								3
V								3
Z								3
Ž								3



21 pav. Lietuvių kalbos gestų abėcėlei atpažinti reikalingas šablonų skaičius

Paveikslėlyje nr. 21 pateikiama kiekvienai lietuvių kalbos gestų abėcėlei atpažinti reikalingas šablonų kiekis. Rezultatuose matoma, kad didžiajai daliai gestų atpažinti užteko tik trijų šablonų. Sudėtingiausia buvo algoritmą išmokyti atskirti raides „E“, „Ė“ nuo „F“, kadangi šie gestai skyrėsi tik vieno piršto padėtimi. Lentelėje nr. 5 matoma, kai įvedus F raidės gestą reikėjo padidinti „E“, bei „Ė“ raidės gestus. Raidės „G“ gestas pradžioje buvo maišomas su „Č“, tačiau padidinus „G“ šablonų skaičių algoritmas pradėjo juos skirti. Taip pat „R“ raidės gestas reikalavo papildomo šablono, jog suprastų skirtumą tarp „R“ raidės gesto nuo „H“.

Žiūrinti į gestus kurie yra atliekami su judesiu daugiausia problemų kėlė raidės Į, bei J gestai. Šie gestai buvo sunkiai atskiriami kadangi juos atliekant nesikeičia rankos forma, tik pozicija erdvėje. Padidinus šablonų kiekį algoritmas pastebėjo jog vieno gesto pozicija po truputi pasislenka žemyn, o kito gesto pozicija ratu. t. y. pradžios bei galo pozicijos yra vienodos. Lentelėje matoma jog šie gestai buvo įvedami toje pačioje iteracijoje, todėl iškart reikalavo dvigubai daugiau duomenų nei kiti gestai.



22 pav. Šablonų skaičiaus augimas

Šiame grafike matome, kad gautas gestų šablonų augimas, didėjant gestų skaičiui, nėra tiesinis. Didėjant gestų skaičiui, vis daugiau atsiranda panašių gestų, kuriuos sistemai yra sunku atskirti. Dėl šios priežasties gestams atpažinti, reikalingas didesnis šablonų skaičius.

5.2. Lietuvių kalbos gestų abėcėlės raidžių atpažinimo tyrimas

5.2.1. Tyrimo tikslas

Antrojo tyrimo tikslas sužinoti, kaip tiksliai sistema geba atpažinti skirtingų vartotojų gestus. Šio tyrimo metu sistema taip pat būtų tobulinama pridodant naujus gestų šablonus, bei sekamas šių gestų šablonų skaičių kitimas. Kurie gestai vartotojui buvo sudėtingi atlikti, bei „Leap Motion“ keliamos problemos taip pat yra labai svarbi informacija. Kiekvienas vartotojas tuos pačius gestus gali atlikti skirtingai. Pavyzdžiui vieni gestą su bangos judesiu gali atlikti plačiais mostais, kiti siauresniais. Tyrimo metu pamačius tokius atvejus, sistemą galima būtų išmokyti atpažinti abejus variantus. Svarbu patikrinti ar šie variantai nesugadina sistemos tikslumo.

5.2.1. Pasiruošimas tyrimui

Sistemos duomenų bazėje saugomi gestų šablonai yra gauti iš pirmojo eksperimento. Kaip ir pirmojo eksperimento metu, ant stalo vartotojui pastatomas „Leap Motion“ įrenginys. Vartotojas turės rodyti gestus, jog rankos plaštaką būtų į „Leap Motion“ prietaisą (20 pav).

5.2.2. Tyrimo eiga

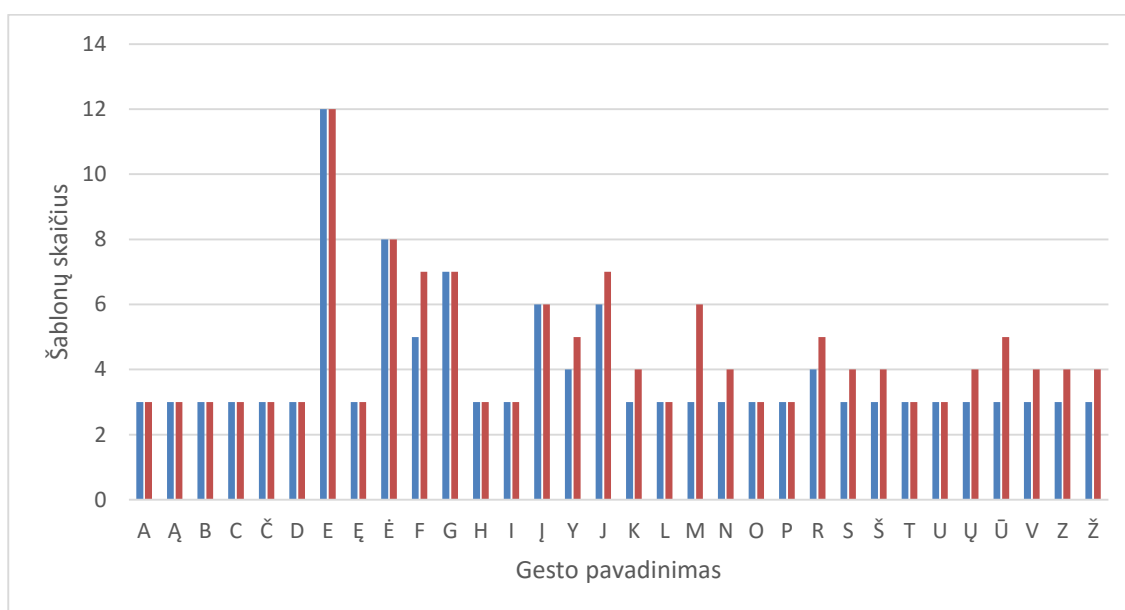
Tyrimo dalyvavo du asmenys. Pirmasis tyrimo dalyvis yra keturiasdešimt septynerių metų moteris, kuri iki šio tyrimo neturėjo jokio patirties naudojantis „Leap Motion“ įrenginiu. Antrasis tyrimo dalyvis yra dvidešimt penkerių metų vyras, kuris turėjo mažos patirties naudojantis „Leap

Motion“ įrenginiu. Abu asmenys yra neutralūs dalyviai kadangi neprisidėjo prie sistemos kūrimo. Prieš tyrimą abu dalyviai apsiprato su „Leap Motion“ įrenginių rodant tam tikrus abėcėlės gestus.

Tyrimo metu dalyvis turėjo parodyti kiekvieną gestą tris kartus. Jei gestas buvo atpažįstamas su klaidomis, buvo bandoma algoritmą išmokyti dalyvio rodomais gestais. Tokiu atveju bandymas to gesto buvo padaromas iš naujo. Jei papildomas šablonų pridėjimas nepadėdavo pasiekti geresnių rezultatų, nauji šablonai būdavo ištrinami. Šie šablonai taip pat būdavo ištrinami, jei testai kurie buvo sukurti pirmojo bandymo metu parodėdavo klaidą.

5.2.3. Tyrimo rezultatai

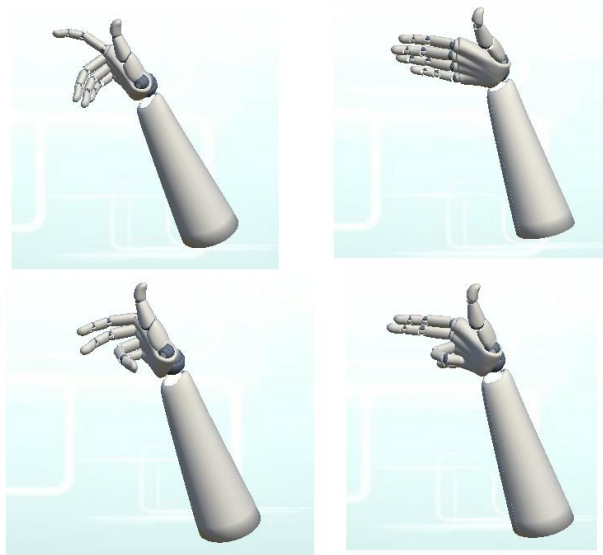
Pirmasis tyrimo dalyvis



23 pav. Šablonų pokytis po tyrimu su pirmu dalyviu

Pirmam asmeniui atlikinėjant bandymus gestų šablonų kiekis keletui raidžių didėjo. Kaip ir ankstesniuose bandymuose bėdas kėlė F raidės gestas, kadangi jis buvo maišomas su E, bei Ė raidės gestais. Norint padidinti tikslumą buvo padidintas F raidės gestas, tačiau daug bėdų kėlė netaisyklingas iš „Leap Motion“ pateikiamas rankos erdvinis modelis. Taip pat buvo bėdų atskiriant I nuo J raidžių gestus, dėl to teko padidinti J raidės gestą.

Nauja bėda kilo su M, bei N raidžių gestais. Šie gestai yra puikiai atpažįstami jei „Leap Motion“ pateikia teisingus duomenis, tačiau dažnai šis įrenginys ranką iškreipdavo (pav. 24). Labiau įgudęs su „Leap Motion“ vartotojas žino kaip rodyti gestą jog jį traktuotu tinkamai, tačiau kitiems vartotojams kyla tos pačios rankų iškreipimo problemos. Apmokius algoritmą su nevysiškai tiksliai gestų šablonais bėda išsisprendė. Šis įvedimas gali kelti bėdų jei į sistemą būtų įvedama naujų gestų, tačiau šiuo atveju problemos nėra.



24 pav. Kairėje netinkamas rankos erdvinis modelis. Dešinėje tinkamas.

Raidžių gestams S, bei Š buvo pridėti papildomi šablonai, kadangi vartotojai nelabai mokėjo juos taisyklingai rodyti. Dėl šios priežasties algoritmas šiuos gestus tarpusavį maišydavo.

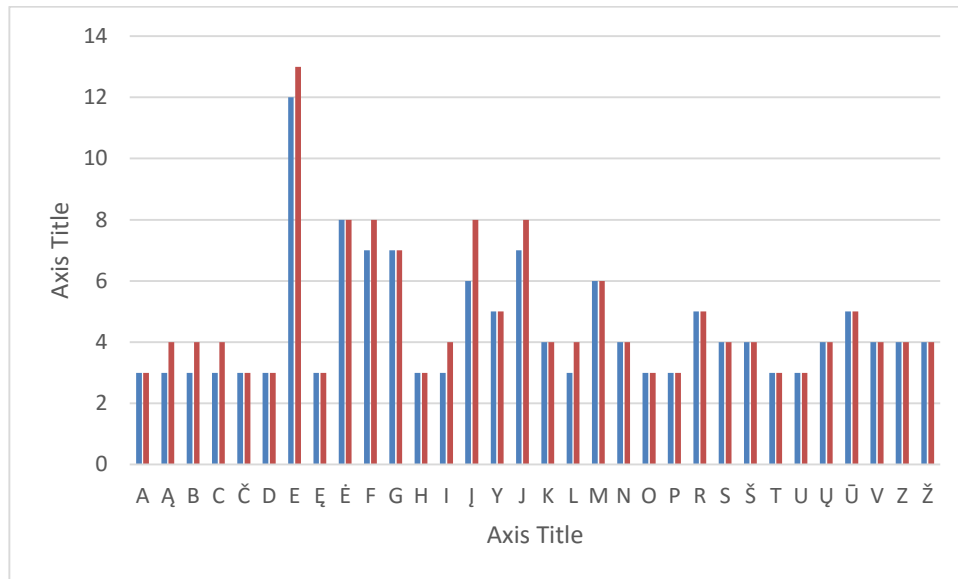
Daugiausiai šiame tyrime kėlė problemų Ū, bei Ų raidžių gestai. Jiems buvo pridėti papildom šablonai, tačiau problema neišsprendė. „Leap Motion“ prietaisas netaisyklingai juos atvaizduodavo. Kadangi šie gestai yra atliekami su judesiu dažnai pasireikšdavo rankos iškraipymo problema. Aiškiai matėsi kad tokių netaisyklingų gestų įvedimas gali sugadinti labai daug judančių gestų atpažinimui, todėl šių gestų šablonų skaičius nebuvo smarkiai didinamas.

Lentelė nr. 6. Tyrimo su pirmu dalyviu rezultatai

Gestas	Pavykusių bandymų skaičius	Komentaras
A	3	
Ą	2 iš 3	Algoritmo problema. Viena kartą neatpažino
B	3	
C	3	
Č	2 iš 3	Algoritmo problema. Užfiksuotas C raidės gestas.
D	2 iš 3	
E	1 iš 3	„Leap Motion“ problema
Ę	1 iš 3	„Leap Motion“ problema
Ė	3	
F	1 iš 3	„Leap Motion“ problema. Buvo bandoma tikslumą padidinti pateikus daugiau šablonų, tačiau tas nepadėjo.
G	3	
H	3	
I	3	
Į	2 iš 3	Algoritmo problemos. Maišydavo su J raidės gestu
Y	3	Tikslumas pasiektas apmokius algoritmą
J	2 iš 3	Algoritmo problemos. Maišydavo su Į raidės gestu. Gesto atpažinimas buvo pagerintas pridėjus daugiau šablonų.
K	3	Tikslumas pasiektas apmokius algoritmą
L	3	
M	3	Tikslumas pasiektas apmokius algoritmą
N	3	Tikslumas pasiektas apmokius algoritmą
O	2 iš 3	Viena karta buvo neatpažintas. Galima problema netaisyklingai atliktas gestas
P	2 iš 3	„Leap Motion“ problema
R	3	Tikslumas pasiektas apmokius algoritmą
S	3	Tikslumas pasiektas apmokius algoritmą
Š	3	Tikslumas pasiektas apmokius algoritmą
T	3	
U	2 iš 3	„Leap Motion“ problema
Ū	1 iš 3	„Leap Motion“ problema
Ů	1 iš 3	„Leap Motion“ problema
V	3	
Z	3	Tikslumas pasiektas apmokius algoritmą
Ž	3	Tikslumas pasiektas apmokius algoritmą

Lentelėje 6 matomas gestų atpažinimo tikslumas. Šis vartotojas buvo ypač nepatyręs naudojantis „Leap Motion“, dėl to šis įrenginys dažnai pateikdavo klaidingą rankos erdvinį modelį. Dauguma problemų buvo išspręstos įvedant papildomus šablonus. Tai galima pastebėti su Y, K, M, N, R, S, Š, Z, Ž raidžių gestais. Sudėtingesnių gestų atpažinimui tokių kaip F, Į, bei J papildomi šablonai padėjo, tačiau maksimalus tikslumas nebuvo pasiektas.

Antrasis tyrimo dalyvis



25 pav. Šablonų pokytis po tyrimu su antru dalyviu

Šiam asmeniui kilo problemų su Ą raidės gestu. Šis gestas dažnai būdavo neatpažįstamas kadangi reikalavo bangos judesio. Tarp šablonų buvo tik mažos bangos judesiai, todėl rodant didesnę algoritmas šio gesto neatpažindavo. Užteko tik vieno papildomo šablono, jog gestas būtų atpažintas.

B, ir C gestai tikriausiai buvo neatpažįstami dėl kreivos nykščio pozicijos. Vartotojas rodant gestą nykštį labai išriesdavo. Šiuos gestu papildžius vienu papildomu šablonu problemą dingo.

Kaip visuomet problemų šiek tiek kėlė E, É, bei F tarpusavio gestų atskyrimas. Pastebėta jog šiuos gestus, o ypač E raidės gestas yra sunkiai atpažįstamas „Leap Motion“ įrenginio, kadangi pirštai yra suglausti. „Leap Motion“ prietaisas nemato kai kurių pirštų, kadangi juos užstoja kiti pirštai. Šis įrenginys pradeda traktuoti kur jie gali būti, o tai lemia labai netaisyklingą rankos erdvinio modelio sugeneravimą, bei netikslius duomenis.

Taip pat daug problemų kėlė I ir J raidžių gestų tarpusavio atpažinimas. Pastebėta jog dažna problema per mažas bangos dydis rodant I raidės gestą. Įvedus I raidės gesto šabloną su didesne banga problema dingo.

Raidžių gestų Ū, Ū, U atpažinimas kaip ir anksčiau kėlė daug problemų, kadangi iš „Leap Motion“ gautas erdvinis modelis nebūdavo tikslus. Jei šablonas buvo rodydamas taisyklingai, algoritmas gestą atpažindavo taisyklingai, todėl algoritmas nebuvo tobulinamas naujais šių gestų šablonais.

Lentelė nr. 7. Tyrimo su antru dalyviu rezultatai

Gestas	Bandymų skaičius	Komentaras
A	3	
Ą	3	Tikslumas pasiektas apmokius algoritmą
B	3	Tikslumas pasiektas apmokius algoritmą
C	3	Tikslumas pasiektas apmokius algoritmą
Č	3	
D	3	
E	3	Tikslumas pasiektas apmokius algoritmą
Ę	1 iš 3	„Leap Motion“ problemos
Ė	2 iš 3	„Leap Motion“ problemos
F	3	Tikslumas pasiektas apmokius algoritmą
G	3	
H	2 iš 3	Algoritmo problemos. Maišė su V raidės gestu
I	3	Tikslumas pasiektas apmokius algoritmą
Į	3	Tikslumas pasiektas apmokius algoritmą
Y	3	
J	3	Tikslumas pasiektas apmokius algoritmą
K	3	
L	3	Tikslumas pasiektas apmokius algoritmą
M	3	
N	3	
O	3	
P	3	
R	3	
S	3	
Š	3	
T	3	
U	2 iš 3	„Leap Motion“ problemos
U̇	1 iš 3	„Leap Motion“ problemos
Ū	1 iš 3	„Leap Motion“ problemos
V	3	
Z	2 iš 3	Algoritmo problemos. Gestas buvo neatpažintas
Ž	3	

Lentelėje 7 matomas gestų atpažinimo tikslumas. Dauguma raidžių gestų buvo papildomai apmokomos norint pasiekti maksimalų tikslumą. Raidžių Ę, Ė, U, U̇, Ū gestai buvo retai atpažįstami teisingai, kadangi „Leap Motion“ įrenginys pateikia blogą erdvinį modelį. Šiems gestams nepadėjo ir papildomas šablonų pridėjimas prie algoritmo mokymo.

5.3. Pangramos rašymas naudojant lietuvių gestų kalbos abėcėlės raidžių gestus

5.3.1. Tyrimo tikslas

Sistemą išmokius atpažinti gestus reikalingas teksto įvedimo tyrimas. Šis tyrimas turi būti atliktas skirtingų asmenų norint sužinoti ar sistema elgiasi vienodai. Pagrindinis tyrimo tikslas sužinoti sistemoje pritaikyto gestų fiksavimo metodo tikslumą, bei keliamas problemas. Taip pat svarbu sužinoti ar vartotojai geba lengvai ir be streso naudotis sistema. Turint šią informaciją sistemos gestų fiksavimas gali būti tobulinamas sklandžiau atpažinti gesto pradžią, bei pabaigą.

5.3.1. Pasiruošimas tyrimui

Sistemos duomenų bazėje saugomi gestų šablonai yra gauti iš pirmojo, bei antrojo eksperimento. Kaip ir pirmojo eksperimento metu, ant stalo vartotojui pastatomas „Leap Motion“ įrenginys. Vartotojas turės rodyti gestus, jog rankos plaštaką būtų į „Leap Motion“ prietaisą (20 pav).

5.3.1. Tyrimo eiga

Tyrimui buvo pasirinkta ši pangrama „Įlinkusi fechtuotojo špaga blykčiodama grėžė apvalų arbūzą“. Tyrimo metu yra fiksuojama kiek laiko užtruko įvesti kiekvienai raidei. Jei sistema pateikdavo ne tą gestą, reikėdavo parodyti raidės ištrynimo gestą, bei kartoti atitinkamos raidės gestą. Laikas nebuvo stabdomas neatitikusiam gestui ištrinti, bei raidės gestui pakartoti. Laikas taip pat nebuvo stabdomas dėl su „Leap Motion“ įrenginių susijusiu nestabilumu, tokių kaip neteisingas rankos modelio atvaizdavimas.

Tyrimo dalyvavo du asmenys, kurie puikiai mokėjo naudotis „Leap Motion“ įrenginiu, bei buvo įgudę naudotis gestų atpažinimo sistema. Prieš tyrimą jie pasipraktikavo rašyti kitus tekstus. Pangramos iki tyrimo pražios dalyviai nebuvo matę.

5.3.2.Rezultatai

Pirmasis tyrimo dalyvis

Lentelė nr. 8. Tyrimo su pirmu dalyviu rezultatai. Teksto įvedimo sparta.

Gestas	laikas sek.	Klaidų sk.	Gestas	laikas sek.	Klaidų sk.
J	11,45	1	B	8,33	
L	6,41		L	5,33	
I	4,89		Y	3,26	
N	8,35		K	7,29	
K	9,54		Č	5,08	
U	20,91	2	I	3,88	
S	9,63		O	5,86	
I	5,12		D	8,78	
			A	8,59	
F	19,15	2	M	9,35	
E	9,73		A	2,34	
C	4,86				
H	14,76		G	7,26	
T	8,04		R	10,98	
U	10,81		Ę	8,58	
O	3,21		Ž	4,61	
T	14,53		Ė	13,13	
O	4,41				
J	8,54		A	3,85	
O	6,23		P	2,93	
			V	7,04	
Š	8,28		A	2,91	
P	4,48		L	2,83	
A	5,11		Ū	46,69	19
G	13,44				
A	2,93		A	4,93	
			R	10,18	
			B	5,33	
			Ū	35,83	9
			Z	4,45	
			Ą	3,08	

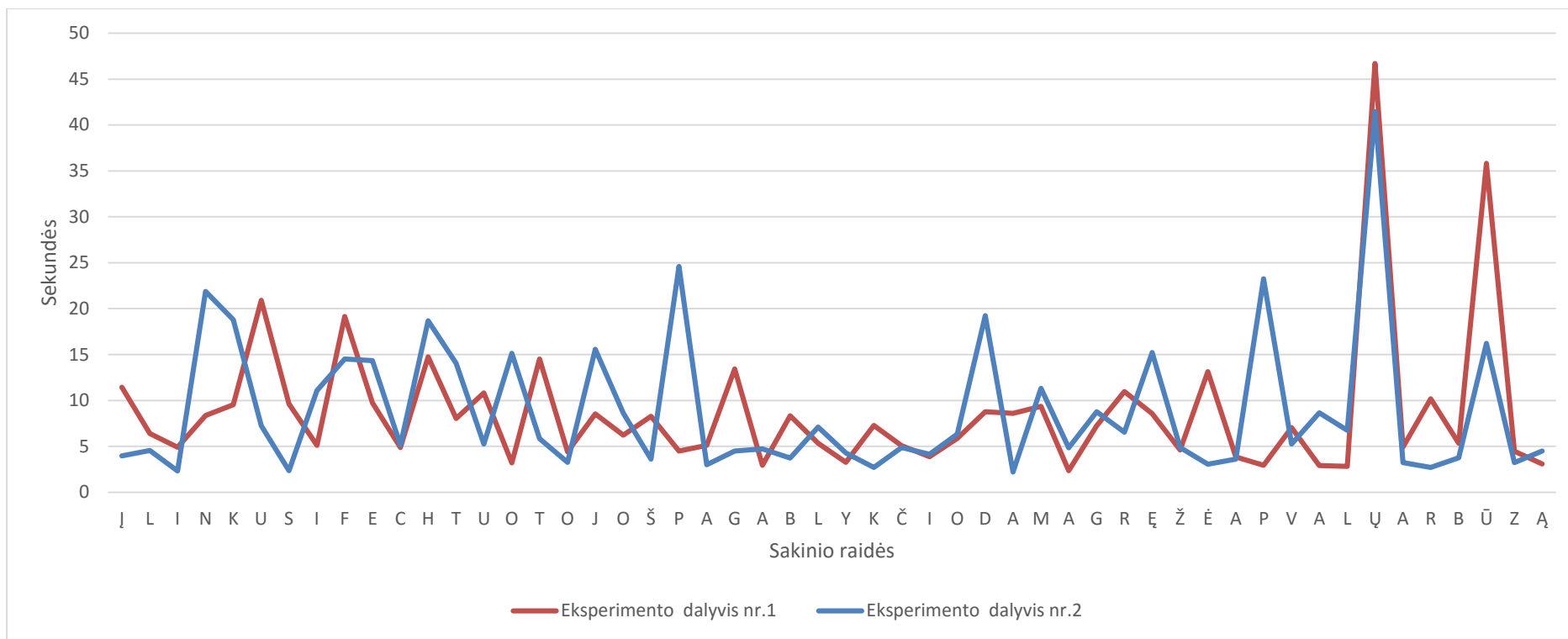
Daugiausia problemų sukėlė Ū, bei Ū raidės gestai. Šie gestai yra atliekami su judesiu, bei „Leap Motion“ įrenginys labai dažnai šiuos gestus atvaizdavo klaidingai. Taip pat sudėtinga buvo atlikti U, bei F raidės gestus, bei su jais padaryta keletą klaidų. Su gestai H, Ė bei T nors ir nebuvo padaryta klaidų, tačiau šiems gestams buvo skirta labai daug laiko atlikti lyginant su kitais gestais. Kiti gestai buvo atpažįstami pakankamai greitai, bei juos atlikti buvo nesudėtinga. Tekstui įvesti užtruko 7 minutes ir 38 sekundes.

Antrasis tyrimo dalyvis

Lentelė nr. 9. Tyrimo su antru dalyviu rezultatai. Teksto įvedimo sparta.

Gestas	laikas sek.	Klaidų sk.	Gestas	laikas sek.	Klaidų sk.
J	3,96		B	3,74	
L	4,55		L	7,1	
I	2,33		Y	4,28	
N	21,86		K	2,7	
K	18,79		Č	4,86	
U	7,24	1	I	4,13	
S	2,35		O	6,33	
I	11,1		D	19,24	1
			A	2,2	
F	14,53	1	M	11,33	
E	14,34	1	A	4,84	
C	5,15				
H	18,68		G	8,76	
T	14,05	1	R	6,54	
U	5,24	1	Ę	15,23	1
O	15,13		Ž	4,86	
T	5,83		Ė	3,04	
O	3,26				
J	15,56		A	3,6	
O	8,61		P	23,26	2
			V	5,24	
Š	3,6		A	8,64	
P	24,59	4	L	6,75	
A	2,99		Ų	41,47	15
G	4,5				
A	4,73		A	3,23	
			R	2,7	
			B	3,76	
			Ū	16,23	3
			Z	3,23	
			Ą	4,48	

Antrajam tyrimo dalyviui kilo problemų su daugiau gestų. Kaip ir pirmuoju atveju labai daug problemų kėlė Ų raidės gestas. Rankos erdvinis modelis buvo atvaizduojamas klaidingai, todėl algoritmas dažnai manė jog P gestas yra E arba F raidės gestas. Atliekant raidžių gestus E, bei Ę taip pat buvo problemų, kadangi jie tarpusavyje buvo maišomi. Ši problema atsirado dėl to, kad tyrimo dalyvis rodydamas E raidę netyčia padarė judesį, dėl to algoritmas šį gesta atpažino kaip Ę. Kiti gestai buvo atpažįstami pakankamai greitai, bei tiksliai. Kai kuriems gestams atlikti užtrukdavo daugiau laiko nei kitus. Tai yra normalu kadangi jie yra su judesiu, arba juos sunku atlikti. Tekstui įvesti užtruko 7 minutes ir 45 sekundes.



26 pav. Pirmojo ir antrojo dalyvio teksto įvedimo spartos sulyginimas

Šiame brėžinyje (26 pav.) pavaizduota kiek užtruko atlikti abejiems tyrimo dalyviams, kiekvienos raidės gestus, pvz. atlikus raidės I gestą buvo užtrukta dvidešimt trys sekundės atlikti raidės N gestui. Lyginant duomenis yra matoma nedidelis skirtumas tarp dviejų vartotojų, tačiau reikėtų įvertinti jog jų naudojimosi su „Leap Motion“ įrenginių įgūdžiai taip pat skiriasi. Antrasis vartotojas bandė gestus atlikti kuo greičiau, todėl padarydavo daugiau klaidų. Pirmasis vartotojas sakinį parašė per septynias minutes, bei trisdešimt aštuonias sekundes. Antrasis per septynias minutes, bei keturiasdešimt penkias sekundes. Skirtumas yra labai nedidelis, tik septynios sekundės.

6. IŠVADOS

Šiame darbe buvo:

1. Išanalizuota rankos gestų atpažinimo naudojant mašininio mokymo algoritmus sritis: apžvelgti taikomi metodai, esamų problemų sprendimo būdai, egzistuojančios sistemos.
2. Suprojektuota ir realizuota gestų atpažinimo sistema paremta mašinių mokymų algoritmais. Sistemą galima išmokyti atpažinti skirtingus gestus papildomai nemodifikuojant sistemos. Sistema buvo išmokyta atpažinti lietuvių kalbos gestų abėcėlės raidžių gestus.
3. Pasiūlytas ir aprašytas iš „Leap Motion“ įrenginio gaunamų duomenų filtravimo metodas. Šiuo metodu apdoroti duomenys yra pateikiami mašinių mokymo algoritmams. Dėl teisingų duomenų šie algoritmai geba greičiau, bei tiksliau atpažinti rodokus getus.
4. Naudojantis sukurta sistema buvo atlikti mašinių mokymo algoritmams reikalingų šablonų kiekių tyrimas, algoritmų tikslumo tyrimas, bei teksto įvedimo greičio rodant lietuvių kalbos gestų abėcėlės raidžių gestus tyrimas.
5. Prieš sistemos kūrimo buvo atliktas tyrimas „Gestų kalbos abėcėlės atpažinimas naudojant Leap Motion įrenginį“, kuris buvo publikuotas „XX-oje magistrantų ir doktorantų konferencijoje informacinės technologijos 2015 m.“

Atliktų eksperimentų išvados

1. Iš antrojo tyrimo rezultatų galima teigti, kad palaipsniui mokant sistemą skirtingų vartotojų gestų šablonais, jos tikslumas gerėja . Iš antrojo dalyvio rezultatų matome, jog sistemai pateikus daugiau skirtingų vartotojų gestų šablonų, buvo atpažinta 25 gestai iš 32, tuo tarpu pirmojo dalyvio su mažiau šablonų tik 19.
2. Ištyrus Leap Motion tikslumą (žr. priede), lietuvių kalbos gestų abėcėlėje buvo nustatyti 19 lengvai atliekamų gestų (kurie viršijo 80 %). Iš šių gestų, sistemai buvo sunku atpažinti tik E raidės gestą. Kadangi sistema gerai sugebėjo atpažinti 25 gestus iš 32 galima teigti, kad sistemoje esantys metodai dalinai išfiltruoja Leap Motion pateikto rankos erdvinio modelio klaidas, bei mašinių mokymo algoritmai gali atpažinti gestus su netikslumais.
3. Iš pirmojo tyrimo matome jog šablonų skaičius, didėjant gestų skaičiui, kinta eksponentiškai. Apmokyti 32 gestus reikėjo 124 gestų šablonų, kur tiesinio augimo atveju turėjo užtekti tik 96. Tai atsitinka dėl didėjančių panašių šablonų skaičiaus. Sistemai juos atskirti darosi sudėtingiau.
4. Teksto įvedimas atliekant lietuvių kalbos gestų abėcėlės raidžių gestus yra pakankamai lėtas, jei naudojamas „Leap Motion“ įrenginys. Trečiojo tyrimo metu nustatyta, kad 52-jų raidžių sakiniui įvesti užtrunka virš septynių minučių, bei padaryta 31 klaida. Literatūros analizės metu gauti duomenys teigia, jog naudojant klaviatūrą įvesti 29-nių simbolių tekstui užtrunka šešias sekundes. Galime daryti išvadą, jog įprastos įvedimo sistemos yra daug kartų greitesnės.
5. Gestų atpažinimas gali būti pritaikytos įvairioms sritims, kurioms įprasto duomenų įvedimo priemonės netinka. Įvertinus atliktų eksperimentinių tyrimų rezultatus galima teigti, kad gestų atpažinimas naudojant mašinių mokymo algoritmus yra vertas dėmesio ir tolimesnių tyrimų bei taikymų.

7. LITERATŪRA

- [1] Hidden Markov model. [Žiūrėta 2016 02 20] prieiga per internetą: https://en.wikipedia.org/wiki/Hidden_Markov_model
- [2] Leap Motion API Overview. [Žiūrėta 2015 02 12] prieiga per internetą: https://developer.leapmotion.com/documentation/csharp/devguide/Leap_Overview.html.
- [3] David Kushner. Oculus Rift Takes Virtual Reality Mainstream. IEEE Spectrum's annual prediction of technologies that will make headlines in the coming year, 2014. [Žiūrėta 2014 10 10]
- [4] VR Team. Oculus Rift [žiūrėta 2016 03 02], prieiga internete: <https://www.oculus.com>
- [5] Leap Motion with Orion [žiūrėta 2016 02 01], prieiga internete <https://www.leapmotion.com/product/vr>
- [6] Leap Motion Gesture Recognition [žiūrėta 2016 04 05], prieiga internete https://developer.leapmotion.com/documentation/csharp/devguide/Leap_Gestures.html
- [7] Pedro Neto, J. Norberto Pires, Member, A. Paulo Moreira. Accelerometer-Based Control of an Industrial Robotic Arm Reality [žiūrėta 2016 02 22], prieiga internete: <https://arxiv.org/ftp/arxiv/papers/1309/1309.2090.pdf>
- [8] UW undergraduate team wins \$10,000 Lemelson-MIT Student Prize for gloves that translate sign language [žiūrėta 2016 05 12], prieiga internete <http://www.washington.edu/news/2016/04/12/uw-undergraduate-team-wins-10000-lemelson-mit-student-prize-for-gloves-that-translate-sign-language/>
- [9] Kamel Boulos et al.: Web GIS in practice X: a Microsoft Kinect natural user interface for Google Earth navigation. International Journal of Health Geographics 2011; prieiga internetu: <http://www.biomedcentral.com/content/pdf/1476-072X-10-45.pdf>
- [10] Keefe, Daniel, Daniel Acevedo, Tomer Moscovich, David Laidlaw, and Joseph LaViola. CavePainting: A Fully Immersive 3D Artistic Medium and Interactive Experience. In the Proceedings of the 2001 Symposium on Interactive 3D Graphics, ACM Press, 85-93, 2001.
- [11] Fakespace Pinch™ Gloves [žiūrėta 2014 11 05], prieiga internete: www.fakespacelabs.com/products/pinch.html
- [12] Kinect Official Web Site. [žiūrėta 2014 11 05], prieiga internete <http://www.xbox.com/kinect/>
- [13] Kinect [žiūrėta 2016 05 05], prieiga internete: <https://en.wikipedia.org/wiki/Kinect>.
- [14] Product Technology: The PrimeSensor™ Technology. [žiūrėta 2014 11 05], prieiga internete: <http://www.primesense.com/?p=487>.
- [15] PrimeSensor™ Reference Design. [žiūrėta 2014 11 05], prieiga internete: <http://www.primesense.com/?p=514>.
- [16] Randy Pausch, Tommy Burnette, University of Virginia; Dan Brockway, Michael E. Weiblen, Science Applications International Corporation: „Navigation and Locomotion in Virtual Worlds via Flight into Hand-Held Miniatures“; Proceedings of the 22nd annual conference on Computer graphics and interactive techniques, Pages 399 – 400, ACM New York, NY, USA ©1995. Prieiga internetu: <http://dl.acm.org/citation.cfm?id=218495>
- [17] Kinected Conference - MIT Media Lab. <http://kinectedconference.media.mit.edu/>.
- [18] DeVincenzi A, Yao L, Ishii H, Raskar R: Kinected conference: augmenting video imaging with calibrated depth and audio. CSCW 2011 Proceedings of the ACM 2011 Conference on Computer Supported Cooperative Work: 19-23 March 2011; Hangzhou, China New York, NY: ACM; 2011, 621-624
- [19] Bi F: Minnesota Prof. Uses Xbox Kinect For Research (The Minnesota Daily, 14 March 2011). [žiūrėta 2014 11 05], prieiga internete: <http://minnesota.cbslocal.com/2011/03/14/minnesota-prof-uses-xbox-kinect-for-research/>.
- [20] Virtopsy - Potential use of gesture control in medicine using the Microsoft Kinect camera (video, 19 December 2010). [žiūrėta 2014 11 05], prieiga internete: [<http://www.youtube.com/watch?v=b6CT-YDChmE>].
- [21] Kreylos O: 3D Video Capture with Kinect (video, 14 November 2010). [žiūrėta 2014 11 05], prieiga internete: <http://www.youtube.com/watch?v=7QrnwoO1-8A>.
- [22] Kreylos O: 2 Kinects 1 Box (video, 28 November 2010). [žiūrėta 2014 11 05], prieiga internete: <http://www.youtube.com/watch?v=5-w7UXCAUJE>.
- [23] Six degrees of freedom [žiūrėta 2016 05 11], prieiga internete https://en.wikipedia.org/wiki/Six_degrees_of_freedom
- [24] Frank Weichert, Daniel Bachmann, Bartholomäus Rudak and Denis Fisseler: Analysis of the Accuracy and Robustness of the Leap Motion Controller (2013); prieiga internete: <http://www.mdpi.com/1424-8220/13/5/6380/htm>
- [25] Sturman, M.M.; Vaillancourt, D.E.; Corcos, D. Effects of aging on the regularity of physiological tremor. J. Neurophysiol. 2005, 93, 3064–3074.
- [26] Burkhard, P.; Langston, J.; Tetrud, J. Voluntarily simulated tremor in normal subjects. Neurophysiol. Clin. 2002, 32, 119–126.
- [27] Accord Framework [žiūrėta 2016 05 11], prieiga internete <http://accord-framework.net/>
- [28] Unity game engine [žiūrėta 2016 05 12], prieiga internete <https://unity3d.com>
- [29] Leap Motion assets [žiūrėta 2016 05 12], prieiga internete <https://developer.leapmotion.com/documentation/unity/index.html/>
- [30] Robertas Sturonas: Leap Motion judesio sensorių panaudojimas naudotojo sąsajų kūrimui; XIX tarpuniversitetinė magistrantų ir doktorantų konferencija „Informacinė visuomenė ir universitetinės studijos“ (IVUS 2014)
- [31] Lietuvių gestų kalbos pirštų abėcėlė. [Žiūrėta 2015 03 29] prieiga per internetą: <http://gestai.unriis.lt/index.php/pirstu-abecele>
- [32] American Sign Language. [Žiūrėta 2015 03 29] prieiga per internetą: <http://www.lifefprint.com/asl101/fingerspelling/fingerspelling.htm>

8. PRIEDAI

GESTŲ KALBOS ABĖCĖLĖS ATPAŽINIMAS NAUDOJANT LEAP MOTION ĮRENGINĮ

Aurelijus Vaitkevičius¹, Mantas Taroza²

¹*Kauno technologijos universitetas, Programos inžinerijos katedra, L. Žemkalnio g. 9-27, Kaunas, Lietuva, aurelijus.vaitkevicius@gmail.com*

²*Kauno technologijos universitetas, Programos inžinerijos katedra, Kovo 11-osios g. 73-98, Kaunas, Lietuva, mantas.taroza@gmail.com*

Santrauka – Šiame straipsnyje yra rašoma apie gestų kalbos abėcėlės atpažinimo sistemas, esamas problemas ir pritaikymo galimybes naudojant Leap Motion įrenginį. Tyrimui pasirinktos amerikiečių ir lietuvių gestų kalbų abėcėlės. Tyrimo metu sulyginami duomenys, t.y. Leap Motion įrenginio sudaromas rankos erdvinis vaizdas su realiais tyrimo dalyvio rankų gestais. Tyrimas parodė, jog daugumoje atvejų gestai yra interpretuojami nekorektiškai.

- **Raktiniai žodžiai:** gestų kalba, Leap Motion, gestų atpažinimas, gestais paremtos sąsajos, gestų atpažinimo tikslumas.

1 Įžanga

Plečiantis vaizdo atpažinimo technologijų gausai, šios technologijos dažniau naudojamos vartotojų sąsajoms valdyti [6]. Jos leidžia spręsti įvairias vartotojo sąsajų problemas, kurios atsiranda naudojant standartinę įrangą, tokią kaip kompiuterinė pelė ar klaviatūra. Nors ši sritis ir joje pasirodžiusios technologijos nebe naujos, vis dar iškyla problemų, kaip tiksliai traktuoti žmogaus gestus. Techninės įrangos trūkumus bandoma spręsti programiniais sprendimais. Dėl to, kompanijos pradėjo specializuotis į tam tikras specifines žmogaus gestų atpažinimo sritis, tokias kaip kūno, rankos ar pirštų judesių atpažinimas.

Vieni pirmųjų sukūrę žmogaus gestais paremtą sąsajos įrenginį, kuris pasiekė didelį komercinį pripažinimą, yra Microsoft Kinect [1]. Šis įrenginys naudoja infraraudonųjų spindulių kamerą gylio nustatymui ir tuo remiantis sudaro žmogaus skeleto erdvinį modelį. Leap Motion technologija naudoja panašų metodą, tačiau jos programinė įranga atpažįsta tik žmogaus rankas ir sudaro jų erdvinį modelį [2].

Leap Motion matymo spektras yra kur kas mažesnis, nei įprastinių vaizdo atpažinimo įrenginių, tačiau jo tikslumas yra aukštas, t.y. 0,2 mm tikslumu [3]. Tai suteikia teorinę galimybę Leap Motion įrenginį naudoti gestų kalbos atpažinimui. Leap Motion programinė įranga atpažįsta keletą rankos gestų, kurie vadinami šabloniniais gestais [4].

Viena iš sričių, kur būtų galima pritaikyti labai tikslų rankų gestų atpažinimą, yra nebylių naudojamos gestų kalbos atpažinimas ir interpretavimas [5]. Šių sistemų įgyvendinimui pasitelkiamos technologijos, tokios kaip Data Gloves, Microsoft Kinect ar vaizdo atpažinimas naudojant paprastas kameras. Leap Motion technologijos pranašumas yra tas, jog ji yra nesudėtinga ir lengvai prieinama vartotojams dėl savo kainos.

Šiame straipsnyje atliekamo tyrimo pagrindinis tikslas yra išsiaiškinti ar dabartinis Leap Motion įrenginio tikslumas, fiksuojant rankų judesius, leistų sukurti virtualios aplinkos sąsają, su kuria sąveikauti būtų galima remiantis gestų kalba.

2 Gestų kalbos atpažinimas naudojant MS Kinect, duomenų pirštines

MS Kinect yra skirtas žmogaus kūno padėties ir judesių fiksavimui atliekant vaizdo analizę ir atpažinimą. Naudojant šį įrenginį yra sukurtos gestų kalbos žodžių atpažinimo sistemos. Žodžių atpažinimas yra atliekamas sekant rankų judėjimo trajektoriją ir padėtį erdvėje. Dažniausiai gesto atpažinimo procesas vyksta sulyginant Kinect įrenginiu gautus duomenis su sistemoje saugomais gestų šabloniniais duomenimis [7,8]. Gestų kalbos abėcėlės raidžių atpažinimas naudojant MS Kinect yra beveik neįmanomas. Šis įrenginys yra pritaikytas platiems gestams atpažinti, kuriems nereikia sekti smulkių žmogaus kūno dalių, t.y. pirštų, rankos pasukimo kampo.

Rankos erdvinį vaizdą galima sudaryti ir technologijomis, kurios nėra paremtos vaizdo atpažinimu. Viena tokių technologijų yra vadinamosios duomenų pirštinės. Jose įmontuoti sensoriai gali sekti pirštų užlinkimus, rankos pasukimą bei padėtį erdvėje [9]. Duomenų pirštinės taip pat naudojamos gestų kalbos atpažinimo sistemoms kurti. Duomenų pirštinės neturi Leap Motion technologijos problemų, kadangi žmogaus rankos bei pirštų pozicijos erdvėje gaunamos iš įrenginyje esančių sensorių. Gaunami duomenys – tikslesni, o erdvė, kurioje atliekami gestai, yra nefiksuota.

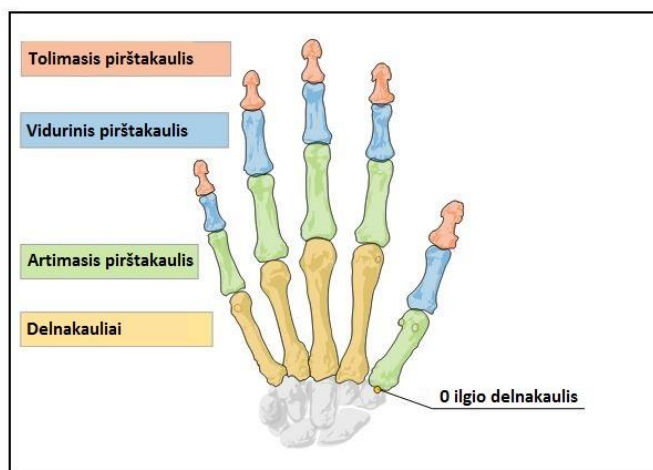
3 Leap Motion veikimo principas

Leap Motion turi dvi monochromines infraraudonųjų spindulių (IS) kameras ir tris IS šviesos diodus. Šiais šviesos diodais yra sugeneruojama 3D IS taškų modelis, kuris yra užregistruojamas monochrominėmis kameromis [3]. Leap Motion technologija yra įspūdinga ne aparatūriniu požiūriu, bet programine įranga. Užregistruotus IS modelius Leap

Motion išanalizuoja matematiniais algoritmais, kurių kompanija iki šiol neatskleidė. Iš dviejų 2D vaizdų, gautų monochrominėmis IS kameromis, Leap Motion sugeneruoja rankų erdvinį modelį. Priešingai nei Microsoft kompanijos kuriamas Kinect, kuris traktuoja pilną žmogaus skeletą, Leap Motion seka tik vartotojų rankas. Įvertinant žmogaus rankos galimybes, Leap Motion programinė įranga gali numanyti pirštų, delno ar riešo padėtis ir tuo atveju, kai kameros šių vietų nemato. Rankas Leap Motion gali traktuoti nuo 25 iki 600 milimetrų atstumu, 150 laipsnių kampu, leidžiant vartotojui laisvai jaustis erdvėje. Nors ir šios charakteristikos yra svarbios, pats didžiausias Leap Motion privalumas yra jo programavimo sąsaja (API*), kuris leidžia lengvai kurti taikomąją programinę įrangą.

4 Leap Motion Aplikacijų Programavimo Sąsaja

Naudojant infraraudonųjų spindulių kameras, galima nustatyti kiekvieno rankos taško koordinates. Norint atpažinti rankų gestus reikia apdoroti labai daug duomenų nustatant rankos dalis (riešą, plaštaką, pirštus). Leap Motion privalumas yra tas, jog šios sudėtingos duomenų analizės funkcijos yra įdiegtos į Leap Motion programinį paketą [4]. Šis paketas naudingas tuo, jog jis iš 3D vaizdo gauna rankų erdvinį skeletą, išanalizuoja ir sudeda į tam tikrus objektus, kuriuose saugoma atitinkama rankos dalies informacija. Leap Motion API turi tris pagrindinius rankos objektus: pilna ranka, delnas, pirštai. Pilnos rankos objektas suteikia informaciją apie rankos poziciją erdvėje, jos ilgį, bei plotį. Delno objekte saugoma informacija apie rankos identiškumą (kairė ar dešinė), poziciją, bei tos rankos pirštų sąrašas. Pati svarbiausia Leap Motion API dalis, reikalinga gestų traktavimui yra pirštų objektas. Šiame objekte saugoma kiekvieno žmogaus rankos piršto pagrindiniai kaulų duomenys. Žmogaus rankos pirštas turi keturis pagrindinius kaulus (jei įskaitysime plaštakos dalį), išskyrus nykštį, kuris turi tik tris. Leap Motion API piršto objekte nykštys kaip ir kiti pirštų objektai turi keturis kaulus, tačiau apie vieną iš jų niekad nepateikiama jokia informacija (1 Pav.).



1 Pav. Leap Motion kaulų skeletas [4]

5 Tyrimas

Siekiant nustatyti Leap Motion pritaikomumą gestų kalbos atpažinimui, atlikti eksperimentiniai tyrimai. Tyrimo rezultatai naudojami įvertinti Leap Motion galimybes naudoti gestų kalbą virtualios aplinkos valdymui. Svarbiausias tyrimo aspektas yra gestų atpažinimo tikimybė.

Eksperimentinio tyrimo dalyviai buvo penki asmenys nuo 20 iki 44 metų amžiaus, turintys įvairių kompiuterinės įrangos naudojimo patirtį. Tyrimo sąlygos: įprastinė kompiuterizuota darbo vieta, Leap Motion įrenginys ant stalo, įprastas kambario apšvietimas. Prieš pradėdant tyrimą, tyrimo dalyvių buvo paprašyta nusiimti žiedus, laikrodžius, kadangi jie gali įtakoti rezultatus.

5.1 Amerikiečių gestų kalbos abėcėlės atpažinimas

Eksperimento dalyviai turėjo atlikti amerikietiškos gestų kalbos abėcėlės raidžių gestus [11]. Kiekvienos raidės gestas buvo atliktas dešimt kartų. Atliktus dalyvių rankų gestus mes užregistravome Leap Motion aplinkoje, bei nufotografavome realiai rankų rodytus gestus. Vėliau šio tyrimo duomenys buvo analizuojami norint patikrinti ar Leap Motion teisingai užfiksavo atliktus gestus.

Išanalizavus duomenis, pateiktus lentelėje (Lentelė Nr. 1), pastebėta, kad yra problemų aptinkant tarpą tarp pirštų. Vieno centimetro tarpai yra blogai traktuojami ir sunku nustatyti kada rodomas C, o kada O gestas.

Gestai, kuriems reikalinga tiksli nykščio padėtis, taip pat yra sunkiai nustatomi. Nykštys yra dažnai uždengiamas kitų pirštų. Nykščio padėtį Leap Motion bando nuspėti. Gaunama apytikslė padėtis, kurios neužtenka tiksliam raidžių E, M, N, T, H, K, S, V, X gestų traktavimui.

* API - application programming interface

Pagrindinės problemos, pastebėtos tyrimo metu, yra tos, jog kai kuriems gestams atpažinti reikia labai tikslaus 3D rankos erdvinio vaizdo. Tai akivaizdžiai pasireiškė atliekant P raidės gestus, kai pirštai yra tik dalinai užlenkti (nesugniaužti į kumštį), tačiau iš delno pusės įrenginys nustatė pirštus kaip pilnai užlenktus. Taip pat ši problema pasireiškė su raidės R gestu [11], t.y. pirštai turi būti sukryžiuoti, bet jie yra pateikiami kaip suglausti, o toks gestas atitinka U raidę [11]. Leap Motion nustato žmogaus rankos erdvinę formą atsižvelgiant į pirštus. Problema pasireiškė, kai rodomas sugniaužtos rankos gestas ir Leap Motion nesugebėdavo tiksliai nustatyti rankos pasukimo kampo ir dėl to atvaizduodavo ją pasuktą neteisingu kampu. Toks įrenginio elgesys taip pat būdingas ir jau aptartais atvejais, kai dalinai užlenkti pirštai traktuojami kaip pilnai užlenkti. Tokiu atveju taip pat ranka yra pasukama, nors toks veiksmas nėra atliekamas.

Lentelė Nr. 1 Leap Motion amerikiečių gestų kalbos raidžių atpažinimo duomenys

Raidė	Atpažinimas,%	Raidė	Atpažinimas,%	Raidė	Atpažinimas,%
A	94	J	70	S	46
B	48	K	44	T	8
C	50	L	80	U	62
D	66	M	10	V	66
E	54	N	0	W	86
F	76	O	70	X	56
G	82	P	56	Y	86
H	66	Q	32	Z	88
I	70	R	46		

Iš lentelėje (Lentelė Nr. 1) pateiktų eksperimentinio bandymo rezultatų matome, jog gautas tikslumas yra netinkamas norint sukurti sąsają, kuri atpažintų nebylių kalbos gestus. Tyrimas atskleidė Leap Motion įrenginio, naudojamo vaizdo analizei, algoritmo spragas, kurios neleido atpažinti dalies gestų. Taip pat pastebėjome kelis gestus, kuriuos Leap Motion puikiai traktuoja, ir kurie šiuo metu gali būti pritaikyti įvairioms vartotojo sąsajoms.

5.2 Lietuvių gestų kalbos abėcėlės atpažinimas

Šios tyrimo dalies metu eksperimento dalyviai turėjo atlikti lietuvių gestų kalbos abėcėlės raidžių gestus [10]. Kiekvienos raidės gestas buvo atliktas dešimt bandymų. Kaip ir ankstesnės tyrimo dalies metu, atlikti gestai buvo užregistruoti ir lyginami su realiai rodomu rankos gestu. Dauguma lietuvių kalbos abėcėlės gestų yra panašūs į amerikiečių kalbos gestus. Rezultatai pateikti lentelėje (Lentelė Nr. 2).

Lentelė Nr. 2 Leap Motion lietuvių gestų kalbos raidžių atpažinimo duomenys

Raidė	Atpažinimas,%	Raidė	Atpažinimas,%	Raidė	Atpažinimas,%	Atpažinimas,%	Raidė
A	100	É	70	K	44	Š	96
Ą	100	F	70	L	80	T	70
B	88	G	88	M	100	U	50
C	30	H	56	N	50	Ū	52
Č	54	I	70	O	100	Ū	70
D	72	Į	90	P	45	V	64
E	88	Y	86	R	46	Z	88
Ę	94	J	92	S	60	Ž	95

Atlikus tyrimą, pastebėtos tos pačios problemos atpažįstant gestus, kurios pasireiškė su amerikiečių gestų kalbos abėcėle. Problemos atsiranda tuomet, kai Leap Motion įrenginys nemato tam tikrų pirštų. Tuomet jų padėčių negalima užfiksuoti ir jie yra identifikuojami netinkamai. Jei pirštai yra prastai matomi, jie yra blogai pateikiami erdvėje.

Lietuvių kalbos abėcėlė yra sudėtingesnė nei amerikiečių, kadangi turi daugiau gestų, kurie reikalauja papildomo judesio [10], pavyzdžiui ą ar ū raidžių gestai. Tokius gestus yra sudėtingiau aptikti, kai jie yra greitai rodomi vienas po kito.

6 Išvados

Amerikiečių ar Lietuvių gestų kalbos raidžių atpažinimo sistemos tikslumas turėtų būti ne mažesnis nei 95%. Naudojant duomenų pirštinių technologiją galima pasiekti reikalingą tikslumą, tačiau šie įrenginiai yra brangūs. Juos reikia dėvėti, dėl to yra mažiau patogūs lyginant su vaizdo atpažinimo technologijomis.

Išanalizavus amerikiečių gestų kalbos abėcėlės atpažinimo tyrimo rezultatus pastebėta, jog nebylių kalbos gestų atpažinimas naudojant Leap Motion technologiją yra nepakankamai tikslus norint šiuos gestus pritaikyti vartotojo sąsajoms. Bendras anglų kalbos raidžių gestų atpažinimas siekia tik 60%.

Lietuvių kalbos gestų abėcėlės atpažinimas naudojant Leap Motion įrenginį yra tikslesnis. Daugumoje atvejų atliekant gestą pirštai yra gerai matomi. Kai kurie gestai atliekami su papildomu judesiu. Juos sunkiau atpažinti, kai jie yra greitai rodomi vienas po kito. Bendras lietuvių kalbos raidžių gestų atpažinimas siekia 73%.

Išskyrėme keletą problemų, kodėl ši technologija šiuo metu negali būti panaudota tokiems gestams atpažinti. Pirmiausiai, jei vartotojo tam tikri pirštai yra IS kamerų nematomi, Leap Motion nuspėdamas jų padėtis padaro klaidų, pavyzdžiui ranka yra vaizduojama su užlenktais pirštais kai jie yra ištiesti. Rankai esant sugniaužtai į kumštį, įrenginys nesugeba tinkamai įvertinti rankos pasukimo kampo. Taip pat, nykščio pozicija, kai jis prispaustas prie delno ar yra tarp kitų pirštų, yra nustatoma blogai ir negali būti patikimai naudojama gesto atpažinimui.

Tolesnių tyrimų tikslas yra gauti Leap Motion įrenginio kamerų matomą vaizdą ir pritaikyti papildomus gestų atpažinimo algoritmus, kurie tiksliau atpažintų gestų kalbą.

Literatūros sąrašas

- [1] Kinect for Windows. [Žiūrėta 2015 02 11] prieiga per internetą: <http://www.microsoft.com/en-us/kinectforwindows/meetkinect/features.aspx>
- [2] **Sturonis R.**, Leap Motion judesio sensorių panaudojimas naudotojo sąsajų kūrimui, Informacinės technologijos 2014, IVUS, 2014, 2029–4832.
- [3] **Weichert F., Bachmann D., Rudak B., Fisseler D.**, Analysis of the Accuracy and Robustness of the Leap Motion Controller, Sensors, MDPI, 2013, 13, 5.
- [4] Leap Motion API Overview. [Žiūrėta 2015 02 12] prieiga per internetą: https://developer.leapmotion.com/documentation/csharp/devguide/Leap_Overview.html.
- [5] **Potter L. E., Araullo J., Carter L.**, The Leap Motion controller: A view on sign language, Proceeding of 25th Australian Computer-Human Interaction Conference: Augmentation, Application, Innovation, Collaboration, 2013, 978-1-4503-2525-7.
- [6] **Bishop T.**, Microsoft: 24M Kinects sold, Xbox Live hits 46M members. [Žiūrėta 2015 02 24] prieiga per internetą: <http://www.geekwire.com/2013/microsoft-24m-kinect-sensors-sold-xbox-live-reaches-46m-members/>
- [7] **Chai X., Li G., Lin Y., Xu Z., Tang Y., Chen X.**, Sing Language Recognition and Translation with Kinect. [Žiūrėta 2015 03 24] prieiga per internetą: http://iip.ict.ac.cn/sites/default/files/publication/2013_FG_xjchai_Sign%20Language%20Recognition%20and%20Translation%20with%20Kinect.pdf
- [8] **Lang S.**, Sing Language Recognition with Kinect. [Žiūrėta 2015 03 24] prieiga per internetą: http://www.inf.fu-berlin.de/inst/ag-ki/rojas_home/documents/Betreute_Arbeiten/Bachelor-Lang.pdf
- [9] **Mauricio O. R., Isreal R. I., Jesus P. C.**, Language interpreter for the deaf core using A virtual reality data-glove, Journal of Theoretical and Applied Information Technology, 2011, 33, 2.
- [10] Lietuvių gestų kalbos pirštų abėcėlė. [Žiūrėta 2015 03 29] prieiga per internetą: <http://gestai.unriis.lt/index.php/pirstu-abecele>
- [11] American Sign Language. [Žiūrėta 2015 03 29] prieiga per internetą: <http://www.lifeprint.com/asl101/fingerspelling/fingerspelling.htm>

Recognition of sign language gestures using Leap Motion device

In this research paper, we analyze recognition of sign language gestures using Leap Motion device. Alphabet of American Sign Language is used to analyze accuracy of gestures recognition. We analyze spatial view of hand recorded by Leap Motion, and compare it to real gestures being performed by test subjects. Our research showed that in majority of cases gestures are not recognized correctly.