

KAUNO TECHNOLOGIJOS UNIVERSITETAS
ELEKTROS IR ELEKTRONIKOS FAKULTETAS

Justinas Urbaitis

KVĖPAVIMO KAUKIŲ KLASIFIKAVIMO METODO
SUKŪRIMAS IR TYRIMAS

Baigiamasis magistro projektas

Vadovas
Doc. Dr. Vidas Raudonis

KAUNAS, 2015

KAUNO TECHNOLOGIJOS UNIVERSITETAS
ELEKTROS IR ELEKTRONIKOS FAKULTETAS
AUTOMATIKOS KATEDRA

KVĖPAVIMO KAUKIŲ KLASIFIKAVIMO METODO
SUKŪRIMAS IR TYRIMAS

Baigiamasis magistro projektas
Valdymo technologijos (621H66001)

Vadovas

(parašas) Doc. Dr. Vidas Raudonis

(data)

Recenzentas

(parašas) Doc. Dr. Kastytis Ratkevičius

(data)

Projektą atliko

(parašas) Justinas Urbaitis

(data)

KAUNAS, 2015



KAUNO TECHNOLOGIJOS UNIVERSITETAS

Elektros ir Elektronikos fakultetas

(Fakultetas)

Justinas Urbaitis

(Studento vardas, pavardė)

Valdymo technologijos (621H66001)

(Studijų programos pavadinimas, kodas)

Kvėpavimo kaukių klasifikavimo metodo sukūrimas ir tyrimas

AKADEMINIO SAŽININGUMO DEKLARACIJA

20 15 m. Gegužės 27 d.
Kaunas

Patvirtinu, kad mano, **Justino Urbaičio**, baigiamasis projektas tema „Kvėpavimo kaukių klasifikavimo metodo sukūrimas ir 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)

Urbaitis J. Kvėpavimo kaukių klasifikavimo metodo sukūrimas ir tyrimas. Magistro baigiamasis projektas / vadovas doc. dr. Vidas Raudonis; Kauno technologijos universitetas, elektros ir elektronikos fakultetas.

Kaunas, 2015. 59psl.

SANTRAUKA

Šiame darbe yra aprašomas tyrimas, kurio tikslas yra kokybiškas konvejeriu judančios deguonies padavimo kaukės klasifikavimo metodo sukūrimas ir tyrimas. Pirmoje dalyje yra apžvelgiami metodai, kurie yra naudojami eksperimentiniuose tyrimuose. Pasirinkti metodai yra šie: Ypatingųjų taškų gavimui SIFT ir SURF metodai, klasifikavimui: Euklido atstumas, K-NN artimiausių kaimynų klasifikatorius ir DT – sprendimo medžio klasifikatorius. Antrame skyriuje yra aptariama šių metodų pritaikymas tiriamai sistemai. Trečiame skyriuje yra aprašyta eksperimentinė sistema, jos tyrimai bei gauti rezultatai. Paskutiniame skyriuje yra pateikiamos darbo išvados bei rekomendacijos.

Reikšmingi žodžiai: SIFT, SURF, KNN, DT, klasifikavimas.

Urbaitis J. Development and Research of Respiratory Masks Classification Method. Final project of master degree / supervisor doc. dr. Vidas Raudonis; Kaunas University of Technology, Faculty of Electrical and Electronics Engineering, department of Technology Management.

Kaunas, 2015. 59 p.

SUMMARY

The classification method of respiratory mask is presented in master thesis. Presented method detects a mask on moving conveyer and recognize the type of respiratory mask. The state of the art methods are present in the first part of work. The second section discusses the application of these methods to the test system. The results of experimental investigation are given in the third part. SIFT and SURF features and three types of classifiers, i.e., Euclidian distance, K-NN nearest neighbor and decision tree, are experimentally investigated in the work. Final conclusions, recommendations and references are presented on the end of this work.

Keywords: SIFT, SURF, KNN, DT, classification.

TURINYS

	PAVEIKSLŲ SĄRAŠAS.....	7
	LENTELIŲ SĄRAŠAS.....	9
	SĄNTRUMPŲ IR TERMINŲ ŽODYNAS.....	10
	ĮVADAS	11
1	ANALITINĖ DALIS.....	13
1.1	Požymių išskyrimo metodai.....	13
1.1.1	SIFT vaizdo apdorojimo metodas.....	13
1.1.2	SURF vaizdo apdorojimo metodas.....	17
1.2	Klasifikatoriai.....	19
1.2.1	Euklido atstumas.....	19
1.2.2	K-NN artimiausių kaimynų klasifikatorius.....	19
1.2.3	DT – sprendimų medis.....	21
1.3	Skyriaus apibendrinimas.....	22
2	NAUDOTINI KLASIFIKAVIMO ALGORITMAI.....	23
2.1	Duomenų bazės sudarymas.....	23
2.1.1	Duomenų bazės sudarymo algoritmas SIFT metodu.....	25
2.1.2	Duomenų bazės sudarymo algoritmas SURF metodu.....	26
2.2	Euklido atstumu pagrįstas kvėpavimo kaukės atpažinimo algoritmas.....	28
2.3	K-NN artimiausių kaimynų klasifikavimo algoritmas.....	30
2.4	DT sprendimo medžio klasifikavimo algoritmas.....	31
3	KVĖPAVIMO KAUKIŲ KLASIFIKAVIMO ALGORITMO EKSPERIMENTINIAI TYRIMAI.....	34
3.1	Euklido atstumu pagrįsto klasifikavimo algoritmo eksperimentiniai tyrimai.....	36
3.2	K-NN artimiausių kaimynų metodu pagrįstas klasifikavimo algoritmo tyrimai... 3.2.1 Euklido atstumo taisyklė.....	41
3.2.2	„Cityblock“ atstumo taisyklė.....	43
3.2.3	„Cosine“ atstumo taisyklė.....	46
3.2.4	Koreliacijos atstumo taisyklė.....	48
3.2.5	K-NN artimiausių kaimynų metodu pagrįstas klasifikavimo algoritmo skyriaus apibendrinimas.....	50
3.3	DT sprendimo medžiu pagrįsto klasifikavimo algoritmo tyrimai.....	51
3.4	Klaidingai teigiamas, klaidingai neigiamas testas.....	56
4	IŠVADOS IR REZULTATAI.....	57
5	LITERATŪROS SĄRAŠAS.....	59

PAVEIKSLŲ SĄRAŠAS

1. Pav.	Deguonies I, II ir III tipo padavimo kaukės.	11
2. Pav.	Kvėpavimų kaukių gamybos principinė schema.....	12
3. Pav.	Tas pats objekto kampas, skirtingo dydžio nuotraukose.....	14
4. Pav.	Gauso piramidės sudarymas [2].	15
5. Pav.	Lokalaus ekstremumo paieška [2].	15
6. Pav.	SIFT ypatingo taško deskriptoriaus skaičiavimas ir sudarymas [2].....	16
7. Pav.	Dėžučių filtro aproksimacijos pavyzdys.	17
8. Pav.	Ypatingojo taško krypties nustatymas.....	18
9. Pav.	SURF ypatingojo taško deskriptoriaus sudarymas.....	18
10. Pav.	KNN – veikimo principas.	20
11. Pav.	Sprendimo medžio principinė schema.	21
12. Pav.	Pirmoji duomenų bazė 45° kampu.	24
13. Pav.	Antroji duomenų bazė 90° kampu.....	24
14. Pav.	SIFT metodu pagrįstas duomenų bazės sudarymas.....	25
15. Pav.	SIFT algoritmo pagrindu rasti ypatingieji taškai.....	26
16. Pav.	SIFT ypatingųjų taškų skaičiaus priklausomybė nuo nuotraukos dydžio.	26
17. Pav.	SURF metodu pagrįstas duomenų bazės sudarymas.....	27
18. Pav.	SURF algoritmo pagrindu rasti ypatingieji taškai.....	27
19. Pav.	SURF ypatingųjų taškų skaičiaus priklausomybė nuo nuotraukos dydžio.	28
20. Pav.	Klasifikavimo algoritmas pagrįstas Euklido atstumu.....	29
21. Pav.	Euklido atstumu klasifikuojamų kaukių algoritmas MATLAB programoje.....	29
22. Pav.	Klasifikavimo algoritmas pagrįstas artimiausio kaimyno metodu.....	30
23. Pav.	DT – sprendimo medžio klasifikatoriaus veikimo algoritmas.	32
24. Pav.	Testuojamos nuotraukos su A duomenų baze.	34
25. Pav.	Testuojamos nuotraukos su B duomenų baze.	35
26. Pav.	Klasifikavimo algoritmo, naudojant Euklido klasifikatorių su SIFT ypatingųjų požymių išskyrimo metodu ir A duomenų baze, tikslumas.	36
27. Pav.	Klasifikavimo algoritmo, naudojant Euklido klasifikatorių su SIFT ypatingųjų požymių išskyrimo metodu ir B duomenų baze, tikslumas.	37
28. Pav.	Klasifikavimo algoritmo, naudojant Euklido klasifikatorių su SURF ypatingųjų požymių išskyrimo metodu ir A duomenų baze, tikslumas.....	38
29. Pav.	Klasifikavimo algoritmo, naudojant Euklido klasifikatorių su SURF ypatingųjų požymių išskyrimo metodu ir B duomenų baze, tikslumas.	39
30. Pav.	Parametro <i>Threshold</i> nustatymas.	41
31. Pav.	K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant Euklido atstumą, SIFT metodą ir A duomenų bazę.....	41
32. Pav.	K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant Euklido atstumą, SIFT metodą ir B duomenų bazę.....	42
33. Pav.	K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant Euklido atstumą, SURF metodą ir A duomenų bazę.....	42
34. Pav.	K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant Euklido atstumą, SURF metodą ir B duomenų bazę.....	43
35. Pav.	K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant „Cityblock“ atstumą, SIFT metodą ir A duomenų bazę.....	44

36. Pav.	K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant „Cityblock“ atstumą, SIFT metodą ir B duomenų bazę	44
37. Pav.	K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant „Cityblock“ atstumą, SURF metodą ir A duomenų bazę	45
38. Pav.	K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant „Cityblock“ atstumą, SURF metodą ir B duomenų bazę	45
39. Pav.	K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant „Cosine“ atstumą, SIFT metodą ir A duomenų bazę	46
40. Pav.	K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant „Cosine“ atstumą, SIFT metodą ir B duomenų bazę	46
41. Pav.	K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant „Cosine“ atstumą, SURF metodą ir A duomenų bazę	47
42. Pav.	K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant „Cosine“ atstumą, SURF metodą ir B duomenų bazę	47
43. Pav.	K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant Koreliacijos atstumą, SIFT metodą ir A duomenų bazę	48
44. Pav.	K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant Koreliacijos atstumą, SIFT metodą ir B duomenų bazę	48
45. Pav.	K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant Koreliacijos atstumą, SURF metodą ir A duomenų bazę	49
46. Pav.	K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant Koreliacijos atstumą, SURF metodą ir B duomenų bazę	49
47. Pav.	Ribinio slenksčio <i>Threshold</i> nustatymas DT – sprendimo medžio klasifikatoriui.	52
48. Pav.	DT – sprendimų medžio klasifikatoriaus tikslumo tyrimas, naudojant SIFT metodą ir A duomenų bazę.....	52
49. Pav.	DT – sprendimų medžio klasifikatoriaus tikslumo tyrimas, naudojant SIFT metodą ir B duomenų bazę.....	53
50. Pav.	DT – sprendimų medžio klasifikatoriaus tikslumo tyrimas, naudojant SURF metodą ir A duomenų bazę.....	54
51. Pav.	DT – sprendimų medžio klasifikatoriaus tikslumo tyrimas, naudojant SURF metodą ir B duomenų bazę.....	55
52. Pav.	Klaidingai teigiamas, Klaidingai neigiamas testo rezultatai	56

LENTELIŲ SĄRAŠAS

1. lentelė. Tyrimams naudoto kompiuterio parametrai.....	23
2. lentelė. Klasifikavimo algoritmo, pagrįsto Euklido atstumu, greitaveika.....	40
3. lentelė. Algoritmo greitaveika, naudojant SIFT metodą ypatingųjų taškų radimui.....	50
4. lentelė. Algoritmo greitaveika, naudojant SURF metodą ypatingųjų taškų radimui.....	51
5. lentelė. DT – sprendimo medžio greitaveika.....	55
6. lentelė. Tyrimo rezultatų palyginimas.....	58

SANTRUMPŲ IR TERMINŲ ŽODYNAS

RGB – spalvų maišymo sistema, kurioje naudojamos trys, žmogaus akių receptorių atitinkančios spalvos: raudona (**R**ed), žalia (**G**reen) ir mėlyna (**B**lue).

SIFT (angl. Scale Invariant Feature Transform) – skirtumai tarp gretimų mastelinių vaizdo ir Gauso filtro sąsūkos.

SURF (angl. Speeded Up Robust Features) – pagreitinto klaidoms atsparios savybės.

DistRatio – minimalus santykis tarp daugiausiai grupėje surastų porų kiekio iki sekančio daugiausiai surastų porų kiekio grupėje.

Klaidingai teigiamas (angl. True Positive) – nuotrauka, kuri yra priskirta tam tikrai grupei, ir ji priklauso tai grupei, kuriai yra priskirta.

Klaidingai neigiamas (angl. False Positive) – nuotrauka, kuri yra priskirta tam tikrai grupei, ir ji nepriklauso tai grupei, kuriai yra priskirta.

K-NN – (angl. K – Nearest Neighbors algorithm) – artimiausių kaimynų algoritmas

DT – (angl. Decision tree) – sprendimų medis.

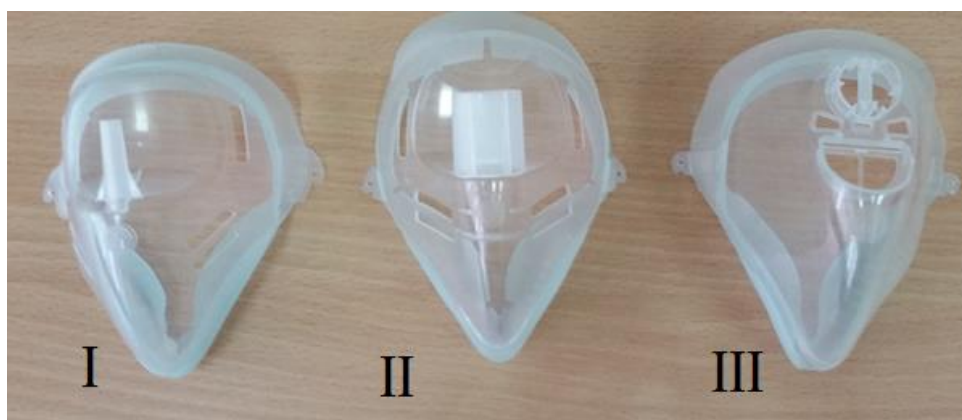
A – Pirmoji duomenų bazė, sudaryta 45° kampu į objektyvą,

B – Antroji duomenų bazė, sudaryta 90° kampu į objektyvą.

Pikselis – (angl. picture (vaizdas) + element (elementas)) tai yra vienas iš daugybės taškų ar vienos spalvos laukelių, kurie, kartu kaip dvimatė matrica, sudaro vaizdą, arba visą nuotraukos turinį.

IVADAS

Šiame darbe apžvelgiamas tyrimo objektas yra konvejeriu judančios deguonies padavimo kaukės, kurios yra naudojamos medicinoje. Jų yra trys rūšys (žr. 1 paveikslą). Jos konvejeriu juda atsitiktine tvarka, padėti bei kampu (žr. 2 paveikslą). Šiuo metu jos yra rūšiuojamos rankomis, todėl kenčia darbo kokybė ir našumas. Siekiant padidinti darbo našumą, nuspręsta sukurti automatizuotą rūšiavimo sistemą, kuri automatiškai nuspręstų, kur kuri kaukė yra, ir jas išrūšiuotų. Tam reikia sukurti kokybišką ir tikslų šių kaukių rūšiavimo algoritmą. Šiame darbe bus apžvelgiami tik kvėpavimo kaukės rūšies atskyrimo viena nuo kitos algoritmo sukūrimai ir tyrimai.



1. Pav. Deguonies I, II ir III tipo padavimo kaukės.

Kaip matyti pirmame paveiksle, jos viena nuo kitos skiriasi nežymiai. Jų formos, dydis ir spalva yra vienodos. Skiriasi tik deguonies padavimo vamzdelių pajungimo tipas, dydis ir vieta.

Kadangi šios deguonies padavimo kaukės viena nuo kitos skiriasi nežymiai, konvejeriu juda atsitiktiniu kampu bei padėti, o su esamomis identifikavimo priemonėmis (pvz.: 3D skaneriais) atskyrimo tikslumas yra gana žemas ir siekia tik ~15%. Tam, kad padidinti atpažinimo tikslumą reikia imtis kitokių priemonių, norint atskirti jas viena nuo kitos. Todėl buvo nuspręsta sukurti greitai veikiančią kompiuterinę kvėpavimo kaukės rūšies atpažinimo algoritmą, kurio dėka atpažinimo tikslumas priartėtų prie ~100%.

Darbo tikslas yra konvejeriu judančių kaukių identifikavimo algoritmo sukūrimas, kurio dėka būtų galima jas sėkmingas atskirti viena nuo kitos. Siekiant greito ir kokybiško rūšiavimo reikia sukurti ir iširti algoritmus, ir parinkti tokį algoritmo derinį, kuris veiktų greičiausiai ir geriausiai.

Šiame darbe bus apžvelgiami ir ištiriami šiuo metu praktikoje naudojami populiariausi metodai vaizdų apdorojime. Bus apžvelgiami požymių išskyrimo metodai bei klasifikatoriai, kurie padės nustatyti reikiamą kvėpavimo kaukės rūšį.

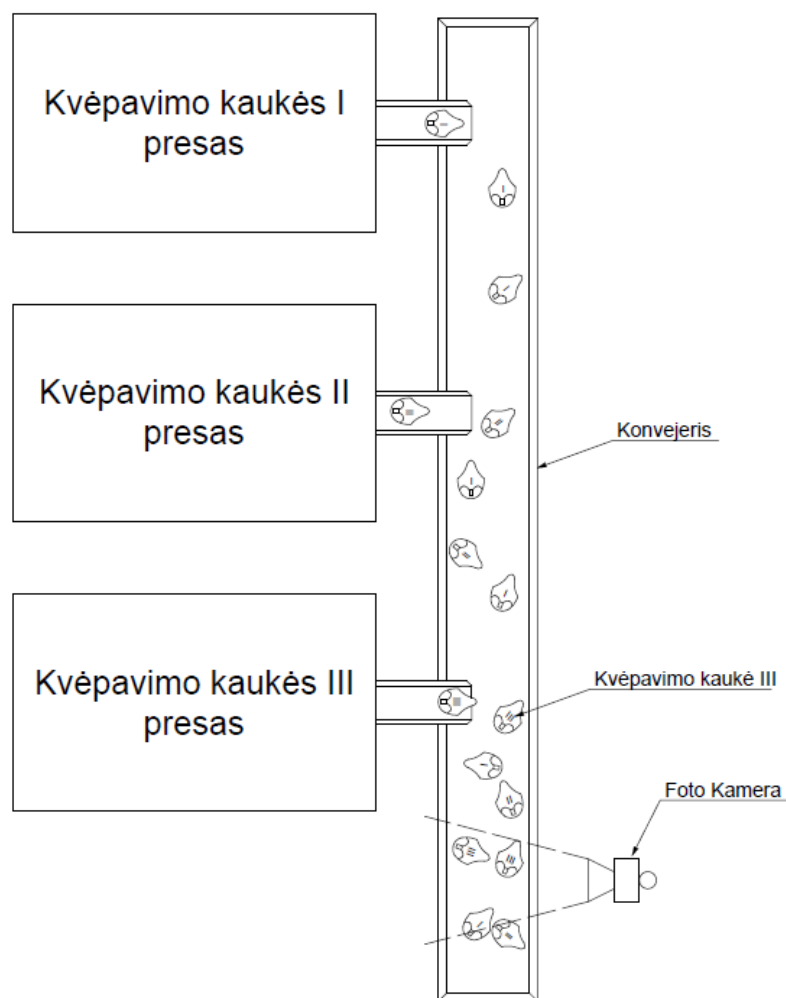
Šiame darbe naudojami požymių išskyrimo metodai yra šie:

- SIFT vaizdo apdorojimo metodas;
- SURF vaizdo apdorojimo metodas.

Šiame darbe naudojami klasifikavimo metodai:

- Euklido atstumas;
- K-NN – artimiausių kaimynų algoritmas;
- DT – sprendimų medis;

Tam, kad pasiekti geriausių rezultatų, visi metodai bus tikrinami su dviem duomenų bazėmis: A – pirmoji duomenų bazė, sudaryta 45° kampu į fotoaparato objektyvą ir B – antroji duomenų bazė, sudaryta 90° kampu į fotoaparato objektyvą. Tai leis adekvačiai įvertinti naudotinus algoritmus, ko pasekoje bus lengviau nustatyti geriausią metodą, tinkantį kvėpavimo kaukės rūšies atskyrimo vienai nuo kitos. Šiame darbe, visi tyrimai yra atliekami su nuotraukomis, kurių dydis yra 800x600 pikselių.



2. Pav. Kvėpavimų kaukių gamybos principinė schema

1 ANALITINĖ DALIS

Požymių aptikimas, jų išskyrimas ir priskyrimas kuriai nors grupei yra dažniausiai kartu derinami metodai, taikomi išspręsti įprastinėms kompiuterinės regos problemoms, tokioms kaip objektų aptikimo ir atpažinimo, veido aptikimo ir atpažinimo tekstūros klasifikavimui ir pan. [1]

1.1 Požymių išskyrimo metodai

Ypatingiesiems taškams surasti ir aprašyti yra sukurta begalė įvairių algoritmų. Vieni iš pirmųjų naudotų metodų buvo linijų radimas „Canny“ metodu [5], kampų lokalizavimas „Harris“ [4] detektoriumi šiuo metu jau yra pasenę ir retai kur naudojami. Pastaruoju metu ieškoma universalių požymių išskyrimo metodų. Taškinių požymių išskyrimo metodams yra keliami universalumo reikalavimai. Jie turėtų veikti su įvairių tipų objektais. Tas pats algoritmas turėtų patikimai veikti bei išskirti pakankamai požymių tiek iš natūralių gamtos nuotraukų, patalpų, įrengimo vaizdų, tiek iš dirbtinai sugeneruotų paveikslų. Taip pat algoritmas turi būti atsparus įvairioms transformacijoms, pavyzdžiui postūmiui, posūkiui, tempimui ir iškraipymui. Be to algoritmas turi būti atsparus apšvietimo pokyčiams, t. y. tiek dieną geru ar blogu oru, tiek esant dirbtiniam apšvietimui turi sutapatinti du taškus iš skirtingų nuotraukų. Algoritmas turėtų būti atsparus žiūrėjimo kampo pasikeitimui.

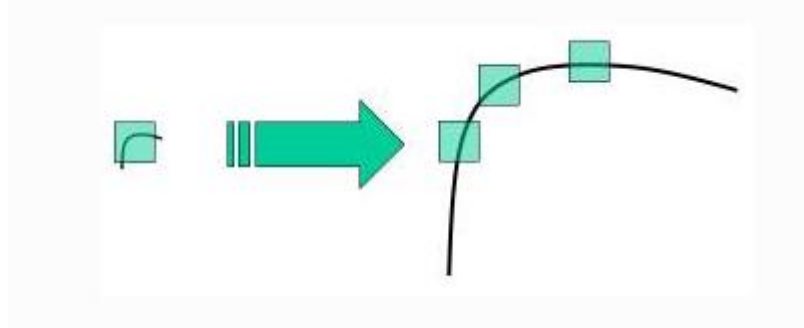
1.1.1 SIFT vaizdo apdorojimo metodas

SIFT yra kompiuterinių vaizdų apdorojimo metodas, skirtas rasti ir apibūdinti nuotraukos ypatinguosius taškus [2]. Kiekvienas daiktas nuotraukoje turi tik jam vienam būdingus ypatinguosius taškus. Šie taškai vėliau gali būti panaudoti identifikuoti objektams kitose nuotraukose tarp daugelio kitų objektų. Naudojant šį metodą yra atliekamas patikimas atpažinimas, net pasikeitus objekto dydžiui, apšvietimui ir posūkiui. SIFT metodu gauti ypatingieji taškai yra atsparūs mastelio ir orientacijos pasikeitimui, bei daliniams nuotraukų iškraipymo ir apšvietimo pasikeitimams.

SIFT metodu pagrįstas algoritmo veikimas apima keturis etapus:

- 1) Dydzio-erdvės ekstremumo radimas;
- 2) Potencialiai ypatingų taškų lokalizavimas;
- 3) Orientacijos priskyrimas;
- 4) Ypatingųjų taškų (deskriptorių) priskyrimas.

Kaip matyti iš 3 paveikslo, negalima teigti, kad objekto kampas kitoje nuotraukoje irgi bus kampas, jei ta nuotrauka yra kitokio dydžio. Todėl pirmame etape yra randami dydžio ir erdvės ekstremumo taškai. Yra akivaizdu, kad negalima naudoti to paties dydžio lango aptikti ypatingiesiems taškams skirtingo dydžio nuotraukose. Todėl tam yra naudojamas dydžio erdvės filtras



3. Pav. Tas pats objekto kampas, skirtingo dydžio nuotraukose.

Vienas iš naudojamų dydžio erdvės filtrų yra aprašomas funkcija: $L(x,y,\sigma)$, kur naudojamas Laplaso - Gauso filtras (angl. Laplacian of Gaussian), kuri yra aprašomas funkcija $G(x,y,\sigma)$. Bendra filtro formulė:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (1)$$

Čia $I(x,y)$ yra nuotrauka, kurioje ieškomi ypatingieji taškai, $*$ yra x ir y sąsūkos operatorius,

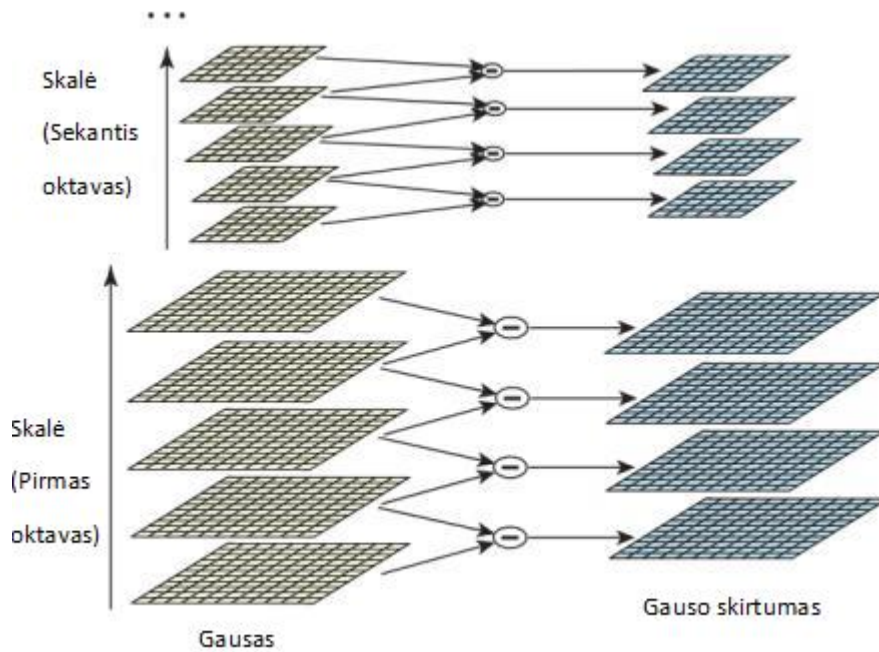
$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \quad (2)$$

kur σ yra Gauso filtro parametras. Pavyzdžiui, antrame paveiksle pavaizduotai nuotraukai, Gauso filtras su mažu σ , tinka labiausiai mažiems kampams, o su didele σ reikšme labiausiai tinka dideliems kampams aptikti. Taip yra aptinkami lokalūs maksimumai arba minimumai nepriklausomai nuo jų dydžio ir erdvės.

Bet šis metodas yra gana lėtas, todėl praktikoje yra dažniausiai naudojamas Gauso skirtumo metodas (angl. DoG - Difference of Gaussian) $D(x, y, \sigma)$, kur DOG yra gautas aproksimuojant LoG funkciją.

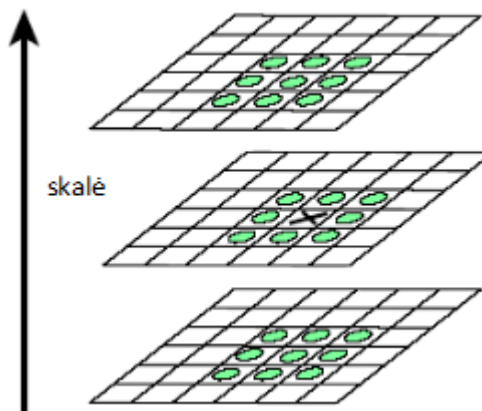
$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (3)$$

Į pirmąją formulę įstačius antrąją gaunamas gauso skirtumas tarp nuotraukos su parametru σ ir sulietos (angl. blurred) tos pačios nuotraukos parametru $k\sigma$. Šis procesas yra atliekamas su įvairiomis oktavomis, ir taip yra gaunama Gauso piramidė (žr. 4 paveikslą). Šis metodas yra gan efektyvus skaičiuojant ypatinguosius taškus.



4. Pav. Gauso piramidės sudarymas [2].

Antrame etape, kai jau yra surasti Gauso skirtumai, ieškoma nuotraukoje lokalaus ekstremumo, visoje nuotraukos skalėje ir erdvėje t. y. kiekvieną nuotraukos pikselį lygina su 8 šalia esančiais pikseliais, taip pats su kitais 9 skalėje esančiais pikseliais, ir prieš tai buvusiais skalėje 9 pikseliais. Ir jei tai yra lokalus ekstremumas, tai jis priskiriamas potencialiam ypatingajam taškui (žr. 5 paveikslą).



5. Pav. Lokalaus ekstremumo paieška [2].

Kai jau yra surasti galimi ypatingieji taškai, jie turi būti tikslinami, tam kad gauti tikslesnius rezultatus. Tam yra naudojamos Teiloro eilutės skalės erdvėje, gauti tikslesnes ekstremumo vietas.

Jei šie ekstremumai yra mažesni už nustatytą slenkstinę ribą (angl. *threshold*) tai jie yra atmetami, ir nepriskiriami ypatingiesiems taškams.

Gauso skirtumas turi didelį atsaką kraštams, todėl kraštus taip pat reikia pašalinti. Tam yra naudojama koncepcija, panaši į Harris kampinio detektoriaus naudojimą. Tam yra naudojama 2x2 Heseno matricos H apskaičiuoti principinį kreivumą.

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \quad (4)$$

Išvestinės yra apskaičiuojamos imant skirtumus tarp kaimynų imties taškų. Ir jei ši vertė yra didesnė nei nustatytas ribinis slenkstis, tai ypatingieji taškai yra atmetami. Tai padeda atmesti mažo kontrasto ir kampinius ypatinguosius taškus. Taip paliekant tik geriausiai atspindinčius nuotrauką ypatinguosius taškus.

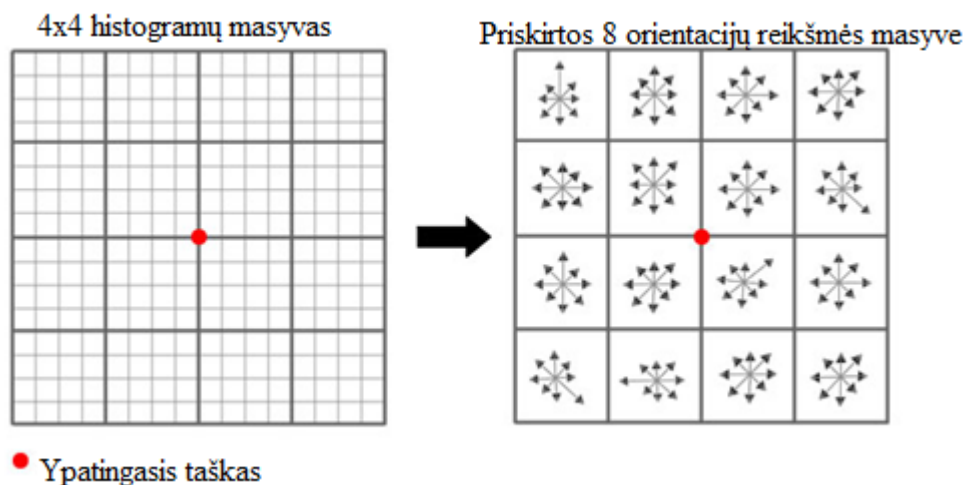
Kai jau yra atmesti nereikšmingi ypatingieji taškai, likusiems ypatingiesiems taškams reikia priskirti lokalią orientaciją erdvėje. Tam yra imami kaimynai aplink ypatingojo taško vietą. Priklausomai nuo ypatingojo taško dydžio gradiento dydis ir kryptis yra apskaičiuojami tame regione:

$$m(x, y) = \sqrt{(L(x + 1, y) - L(x - 1, y))^2 + (L(x, y + 1) - L(x, y - 1))^2} \quad (5)$$

$$\theta(x, y) = \tan^{-1}((L(x, y + 1) - L(x, y - 1)) / (L(x + 1, y) - L(x - 1, y))) \quad (6)$$

Čia $m(x, y)$ yra gradiento vektorius, o $\theta(x, y)$ – gradiento vektoriaus kryptis. Tada sukuriami orientacijos histograma, kurią sudaro 36 laukeliai, kuri padengia 360° laipsnių gradiento orientaciją erdvėje. Taip yra sukuriami ypatingieji taškai, su tam tikra padėtimi ir dydžiu, bet skirtinga kryptimi, taip sukuriant stabilius ypatinguosius taškus.

Ketvirtame etape yra apskaičiuojami ypatingųjų taškų deskriptoriai (žr. 6 paveikslą).



6. Pav. SIFT ypatingo taško deskriptoriaus skaičiavimas ir sudarymas [2]

Deskriptorius yra formuojamas kaip vektorius iš visų sukurtų histogramų įrašų verčių, kur jų dydį ir kryptį atstoja rodyklės kryptis ir ilgis. Taip yra sukuriamas 4x4 histogramų masyvas, kur

kiekviename langelyje yra 8 orientacijų reikšmės. Taip yra sukuriamas 128 ilgio vektorius, kuris yra ypatingojo taško deskriptorius.

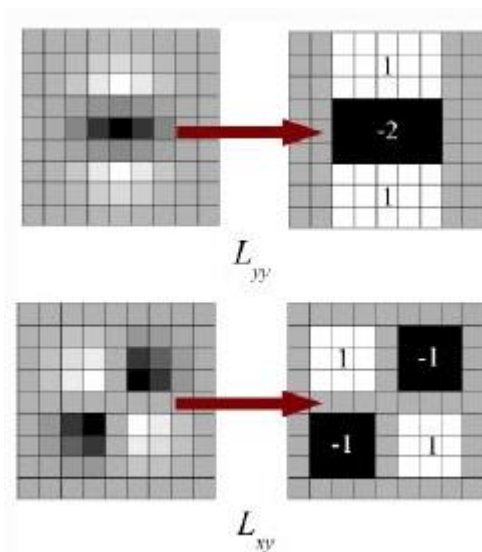
1.1.2 SURF vaizdo apdorojimo metodas

Kompiuterinėje regoje, SURF – pagreitinintas robastiškas pastovių ypatingųjų taškų radimas (angl. Speeded-Up Robust Features) yra vaizdo apdorojimo metodas, skirtas surasti ypatinguosius taškus nuotraukoje [3]. Šis metodas gali būti naudojamas objektų atpažinimui, sekimui, atitikmens ar klastočių radime ar trimačių objektų atkūrimo [6]. Naudojant šį metodą, ypatingieji taškai iš nuotraukos yra išgaunami greičiau nei naudojant SIFT metodą, tai atsispindi ir metodo pavadinime „Pagreitinta SIFT algoritmo versija“

Naudojant SIFT yra aproksimuojamas LoG su DoG filtru, tam kad rasti ypatinguosius taškus. Tačiau SURF metode, šis veiksmas atliekamas kitaip, t. y. LoG yra aproksimuojamas naudojant dėžučių (angl. BOX) filtrą (žr. 7 paveikslą) naudojant Hesiano matricą:

$$H(x, \sigma) = \begin{bmatrix} L_{xx}(x, \sigma) & L_{xy}(x, \sigma) \\ L_{xy}(x, \sigma) & L_{tt}(x, \sigma) \end{bmatrix}, \quad (7)$$

kur $H(x, \sigma)$ – Hesiano matrica duotame taške $x = (x, y)$, $L_{xx}(x, \sigma)$ – Gauso antros eilės išvestinė

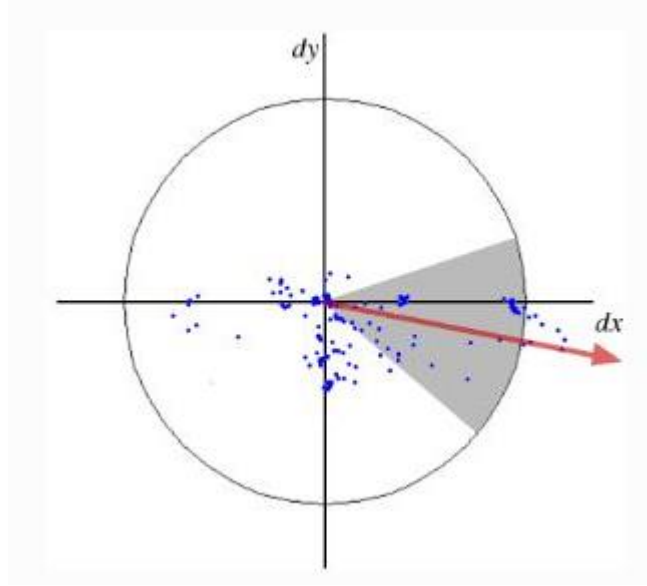


7. Pav. Dėžučių filtro aproksimacijos pavyzdys.

Didžiausias privalumas naudojant šią aproksimaciją yra tas, kad kampų sąsūkos naudojant šį filtrą gali būti lengvai apskaičiuojamos naudojant integralinį vaizdą. Kas taip pat gali būti atliekama ir su skirtingo dydžio nuotraukomis.

SURF ypatingųjų taškų orientacijos priskyrimui naudoja bangelių atsako metodą horizontalia ir vertikalia kryptimi, lyginant su šalia esančiais kaimynais 6s spinduliu nuo ypatingojo taško. Tada

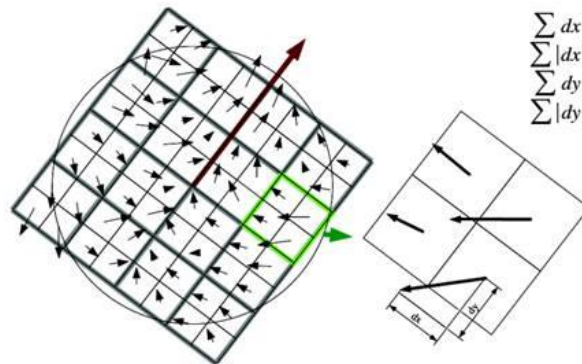
yra atidedami Gauso filtro svoriai, ir jie atidedami erdvėje (žr. 8 paveikslą). Tada yra skaičiuojama dominuojanti orientacija, sumuojant visas gradiento sumas, pasukant jas erdvėje per 60°.



8. Pav. Ypatingojo taško krypties nustatymas

Taip naudojant bangelių transformaciją, su šiuo metodu galima lengvai surasti didžiausią gradiento kryptį, nepriklausomai nuo nuotraukos dydžio. Be to, daugelyje praktikos uždavinių, nereikia surasti ypatingųjų taškų orientacijos, todėl tai pagreitina ypatingųjų taškų skaičiavimą (angl. Speed-UP).

Ypatingųjų taškų deskriptorius yra sudaromas išrenkant taškus 20x20s apie ypatingąjį tašką. Tada šis regionas yra padalinamas į 4x4 dalis (žr. 8 paveikslą).



9. Pav. SURF ypatingojo taško deskriptoriaus sudarymas.

Kiekvienos dalies vertikalūs ir horizontalūs bangelių atsako dydžiai yra paimami ir suformuojamas ypatingojo taško deskriptoriaus vektorius:

$$v = (\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|) \quad (8)$$

Taip yra gaunamas ypatingojo taško deskriptoriaus vektorius, kuris yra sudarytas iš 64 dimensijų. Kuo mažesnis deskriptoriaus ilgis, tuo greičiau vyksta skaičiavimai, bet tai sumažina ypatingųjų taškų tikslumą.

1.2 Klasifikatoriai

Po ypatingųjų taškų suradimo, ir deskriptorių gavimo, juos reikia apdoroti tam, kad būtų galima pasakyti, kuriai klasei priklauso vienas objektas ar kitas, mūsų atveju kuri kaukės rūšis yra tiriama. Šio tipo algoritmai priskiria duomenis tam tikrai klasei. Šiame magistriniame darbe yra apžvelgiami ir ištiriami 3 klasifikavimo metodai:

- Euklido atstumas;
- K-NN – artimiausių kaimynų algoritmas;
- DT – sprendimų medis;

1.2.1 Euklido atstumas

Tai yra klasikinis metodas naudojamas matematikoje. Euklido atstumas[10] – tai yra atstumas tarp dviejų taškų erdvėje. Norint naudoti euklidinę geometriją, reikia turėti du vektorius erdvėje su savo ilgiu $p=(p_1,p_2,\dots,p_n)$ ir kryptimi $q=(q_1,q_2,\dots,q_n)$ ir gaunamas atstumas(d) tarp jų:

$$d(p, q) = d(q, p) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \dots + (q_n - p_n)^2} \quad (9)$$

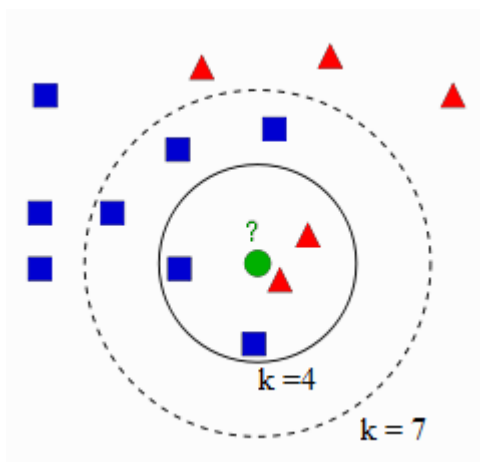
Supaprastinus šią formulę gauname:

$$d(p, q) = d(q, p) = \sqrt{\sum_{i=1}^n (q_i - p_i)^2} \quad (10)$$

Šis atstumas yra vadinamas Euklido atstumu, kuris reiškia atstumą erdvėje tarp dviejų vektorių galų.

1.2.2 K-NN artimiausių kaimynų klasifikatorius.

KNN [7][8] yra klasifikavimo algoritmas, skirtas surasti artimiausiai atitinkantį deskriptorių iš mokymo medžiagos ir priskirti jį kuriai nors klasei (žr. 10 paveikslą). Kaip matyti paveiksle, jame yra dvi klasės: pirma klasė mėlyni kvadratai, o antra klasė raudoni trikampiai. Jie yra atvaizduoti vektorių erdvėje.



10. Pav. KNN – veikimo principas.

Į šią erdvę patalpinus naują kintamąjį – žalią apskritimą, jis užima vietą erdvėje. Šis naujas vektorius turi būti pridėtas prie tam tikros klasės – mėlynos arba raudonos. Šiuo atveju tai yra – klasifikavimas.

Tada yra tikrinama, kur yra artimiausias kaimynas (angl. Nearest Neighbour). Mūsų pavyzdyje, žalias rutuliukas yra arčiausiai raudono trikampio, todėl yra priskiriamas šiai klasei.

Bet kas jei aplink šį tašką yra daug mėlynų kvadratų? Tada mėlyni kvadratai turi daug didesnę įtaką, nei raudoni trikampiai toje erdvės vietoje, kur yra žalias apskritimas. Todėl patikrinti, vien tik artimiausia yra klaidinga, ko pasekoje algoritmas veiks prastai. Todėl yra tikrinama k artimiausių kaimynų. Tada tikrinamas deskriptorius priklausys tai klasei, kuri turės didesnę įtaką toje erdvėje.

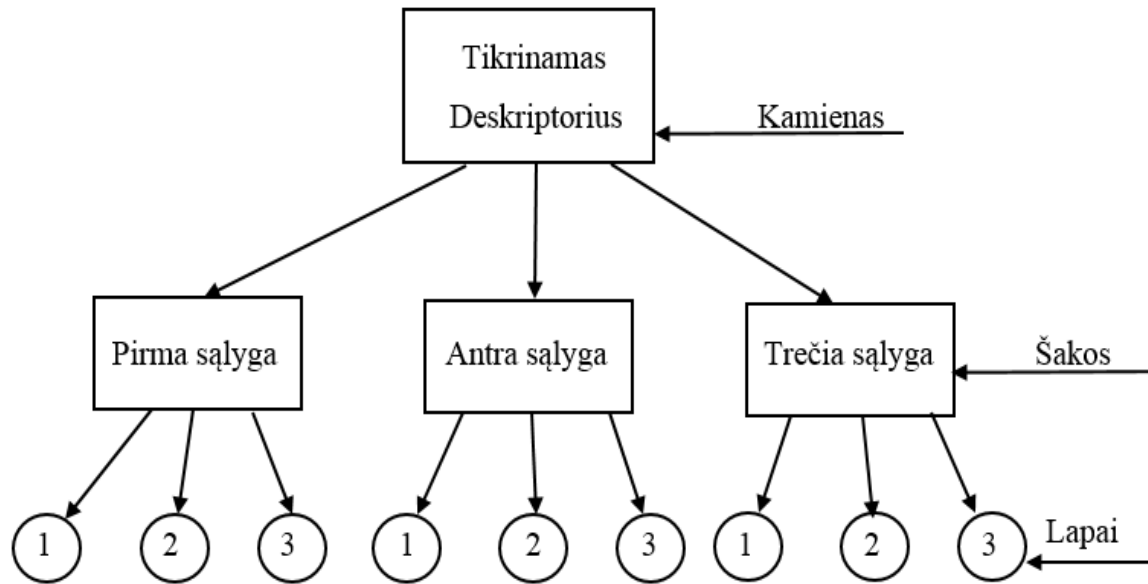
Pavyzdžiui, jei paimama $k = 4$ artimiausius kaimynus. Tada atsiranda du mėlyni kvadratai ir du raudoni trikampiai. Šiuo atveju toje erdvės vietoje yra po 2 kaimynus, todėl įvyksta lygybė ir tikrinamas deskriptorius nėra priskiriamas nei vienai klasei. Bet jei paimama $k = 7$, tada tikrinamoje erdvėje atsiranda 5 mėlyni kvadratai ir 2 raudoni trikampiai. Todėl šiuo atveju didesnę svertą turi mėlyna klasė ir bus priskirta šiai klasei.

Atstumą tarp artimiausių klasių ir tikrinamų nuotraukų galima apskaičiuoti įvairiais būdais. Šiame magistriniame darbe, atstumas tarp klasių ir tikrinamų deskriptorių yra apskaičiuojamas šiais metodais:

- Euklido atstumas
- Citybloc atstumas[9];
- Cosine atstumas;
- Corelation atstumas.

1.2.3 DT – sprendimų medis.

Sprendimų medžiu[12] (angl. Decision tree) yra vadinamas medžio pavidalo klasifikatorius, kuris priskiria tam tikrai klasei ypatingųjų taškų deskriptorius. Sprendimo medžio modelį dažniausiai sudaro medžio, kamienas, šakos, kur yra apima dvi ar daugiau galimų baigčių, lapai, kurie atitinka priskirtą baigtį - klasę, pan. (žr. 11 paveikslą). [13] Kiekviena šaka yra apibrėžiama testo pavidalu, o lapai atitinka visas įmanomas baigtis. Pagrindinė užduotis, naudojant šį klasifikatorių, yra sukurti gerą medžio struktūrą, tam kad būtų galima priimti tinkamus sprendimus. Tam yra reikalinga duomenų bazė, su tiksliai apibrėžtomis klasėmis..



11. Pav. Sprendimo medžio principinė schema.

Pirmiausia deskriptorius paduodamas į medžio kamieną. Jei nagrinėjamas deskriptorius iš karto priklauso kuriai nors baigčiai (lapui), tai jis iš karto yra priskiriamas šiai klasei. Tačiau, jeigu tiriamas deskriptorius nepriklauso kuriai nors klasei (mūsų atveju jų yra trys), tai sukuriamas sprendimo priėmimo mazgo – šakos atsišakojimas. Tada tiriamajam deskriptoriui yra pritaikomos taisyklės, kurios yra gautos po medžio apmokymo [11]. Atlikus skaičiavimus, pagal gautus rezultatus, tikrinamasis deskriptorius yra nukreipiamas vienu lygiu toliau, į kurią nors šaką tolyn. Taip šis procesas yra kartojamas tol, kol yra pasiekiamas galutinis rezultatas, ir tiriamas deskriptorius yra priskiriamas atitinkam lapui – vienai iš trijų klasių.

Sprendimo medžio mokymasis vyksta iš viršaus žemyn, konstravimo principu. Mokymosi tikslas, naudojant duomenų bazę, su iš anksto žinoma priskirta klase, nustatyti klasifikavimo taisykles, kurios tiktų mokymui naudotus deskriptorius, tiek naujai testuojamus deskriptorius. Jeigu mokymo bazėje, yra mažai panašių deskriptorių, tai tada yra sukuriamas lapas, su atitinkama klase, o jeigu

panašių deskriptorių yra daug, tai sukuriama atskira šaka, su tam tikromis taisyklėmis, kurios geriausiai atspindi šiuos deskriptorius. Taip šis procesas yra kartojamas, kol nebelieka mokymosi duomenų bazėje nepriskirtų deskriptorių kuriai nors klasei (lapams).

Kai sprendimų medis jau yra sukurtas, dažniausiai medis būna labai didelis, ir prisitaikęs prie mokymosi duomenų bazės. Ko pasekoje sumažėja medžio tikslumas, ir jis nebegali teisingai priimti sprendimų. Sprendimų medis tampa persimokęs. Todėl šiame darbe naudojamas nereikalingų medžio šakų ir lapų atmetimas – genėjimas (angl. *pruning*). Atliekant šį procesą, po medžio mokymo, arba medžio mokymo metu, kai kurios medžio šakos yra pakeičiamos lapais, tam kad pagerinti priskyrimą kuriai nors klasei. Atlikus šį procesą yra gaunamas didesnis tikslumas naujiems tiriamiesiems deskriptoriams, tuo pačiu mažinant medžio sudėtingumą, kas įtakoja sprendimo medžio greitaveiką.

1.3 Skyriaus apibendrinimas

Šiame skyriuje buvo apžvelgiami metodai, kurie dažniausiai yra naudojami praktikoje. Ypatingųjų taškų išskyrimui buvo pasirinkti du metodai: SIFT dėl savo didelio tikslumo, ir atsparumo pašaliniam poveikiams, bei SIFT, dėl savo greitaveikos ir atsparumo trikdžiams. Kvėpavimo kaukių klasifikavimui pasirinkta: Euklido atstumas, dėl savo paprastumo ir lengvo pritaikymo praktikoje, K-NN artimiausių kaimynų algoritmas – dėl tikslumo ir savo klasifikavimo ypatumų bei DT sprendimų medis – dėl savo greitaveikos, tikslumo ir kitokių apsimokymų principų.

Sekančiuose skyriuose yra aptariama šių metodų pritaikymas konvejeriu judančioms kvėpavimo kaukėms bei gauti rezultatai.

2 NAUDOTINI KLASIFIKAVIMO ALGORITMAI

Šiame skyriuje yra pateikiami algoritmai, kurie yra naudojami kvėpavimo kaukės rūšiai atskirti. Taip pats šiame skyriuje yra apžvelgiama duomenų bazė, bei jos sudarymas. Visi algoritmai yra realizuojami su MATLAB programine įranga. Pirmoje lentelėje yra pateikti kompiuterio parametrai, su kuriais buvo atlikti skaičiavimai:

1. lentelė. Tyrimams naudoto kompiuterio parametrai.

Gamintojas:	DELL
Ekranas	15.4" LCD
Procesorius	Intel® Core™ i7-2670QM CPU 2.20 Ghz
Operatyvioji atmintis	8GB RAM
Kietasis diskas	500GB
Vaizdo plokštė	NVIDIA GeForce GT 540M

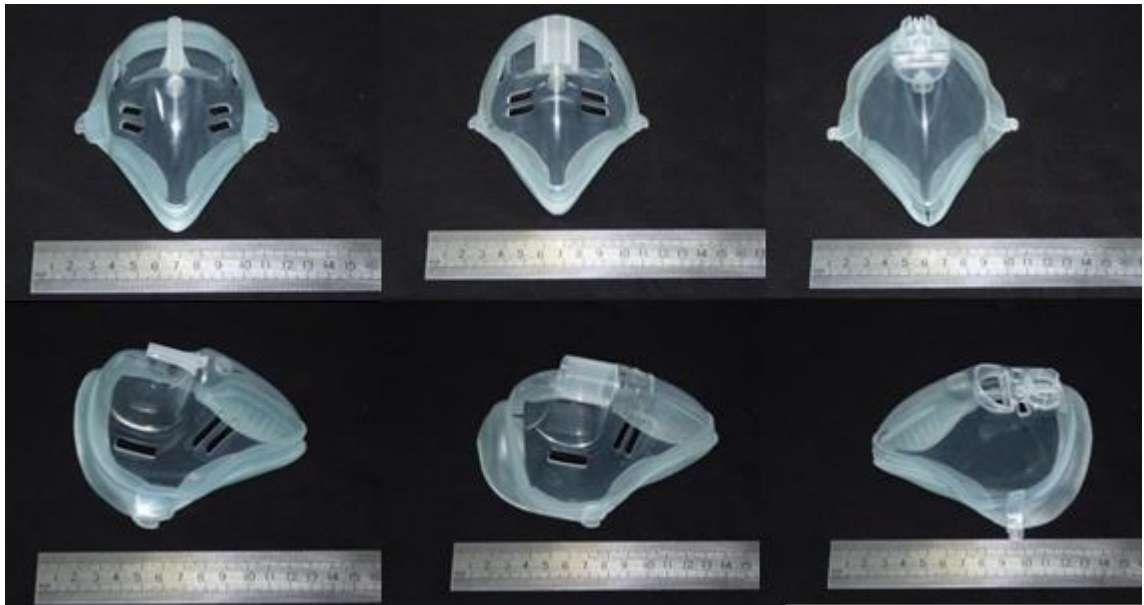
Kvėpavimo kaukės tyrimams analizuoti yra taikomi skaitmeniniai vaizdai, gauti iš fotoaparato kameros, turinčio 14 mln. efektyvių taškų.

2.1 Duomenų bazės sudarymas

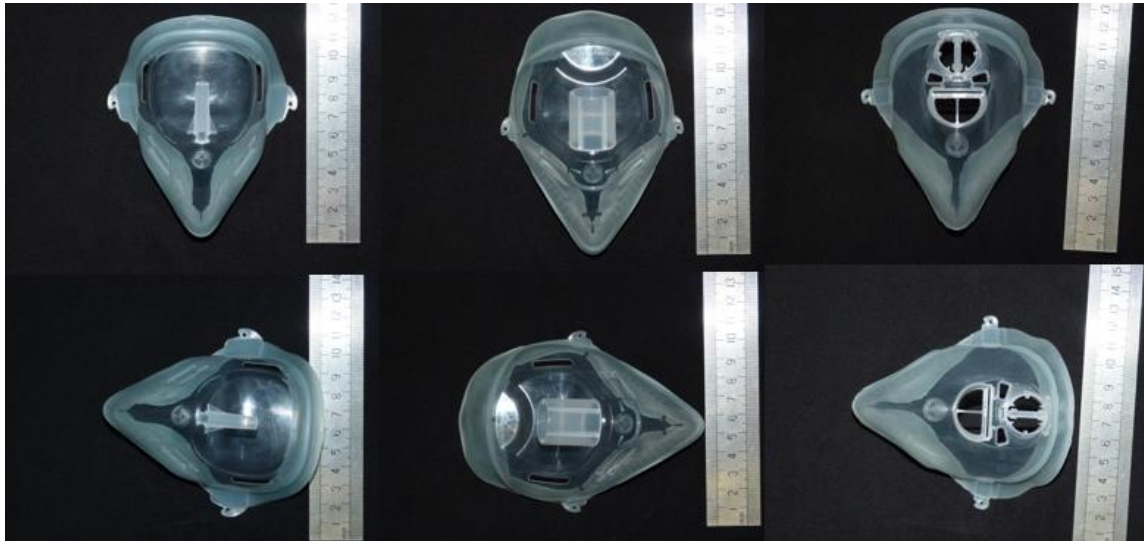
Šio magistrinio darbo tyrimai yra atliekami su dviem duomenų bazėmis, kurių imtis yra apie 300 nuotraukų. Šios duomenų bazės yra sudarytos vienodomis sąlygomis, t. y. esant vienodam apšvietimui, fonui, atstumui, su tomis pačiomis deguonies padavimo kaukėmis, bei tuo pačiu fotoaparatu. Pirmą duomenų bazę yra sudaryta iš nuotraukų su kvėpavimo kaukėmis, kurios į fotoaparato objektyvą yra nukreiptos 45° kampu (žr. 12 paveikslą). Antroji duomenų bazė yra sudaryta iš nuotraukų, kurios į fotoaparato objektyvą yra nukreiptos 90° kampu (žr. 13 paveikslą). Tai yra atliekama tam, kad išsiaiškinti, kokiomis sąlygomis esant, algoritmai veikia geriausiai, bei tam, kad pasiekti geriausius įmanomus rezultatus, bei įvertinti algoritmo veikimą esant neidealiomis sąlygomis.

Antrame skyriuje aprašyti ypatingųjų taškų gavimo metodai naudoja pilkų atspalvių nuotraukas, todėl prieš pradėdant darbus, jas konvertuoti į pilkų atspalvių skalę. Taip yra pašalinama nereikalinga informacija iš nuotraukų, ko pasekoje padidėja nuotraukų apdorojimo greitis. Iš RGB spalvų perėjimas į pilkų atspalvių skalę (1-255 vertės skaičius) įvykdomas taip:

$$Grey = a*R+b*G+c*B \quad (11)$$



12. Pav. Pirmoji duomenų bazė 45° kampu.



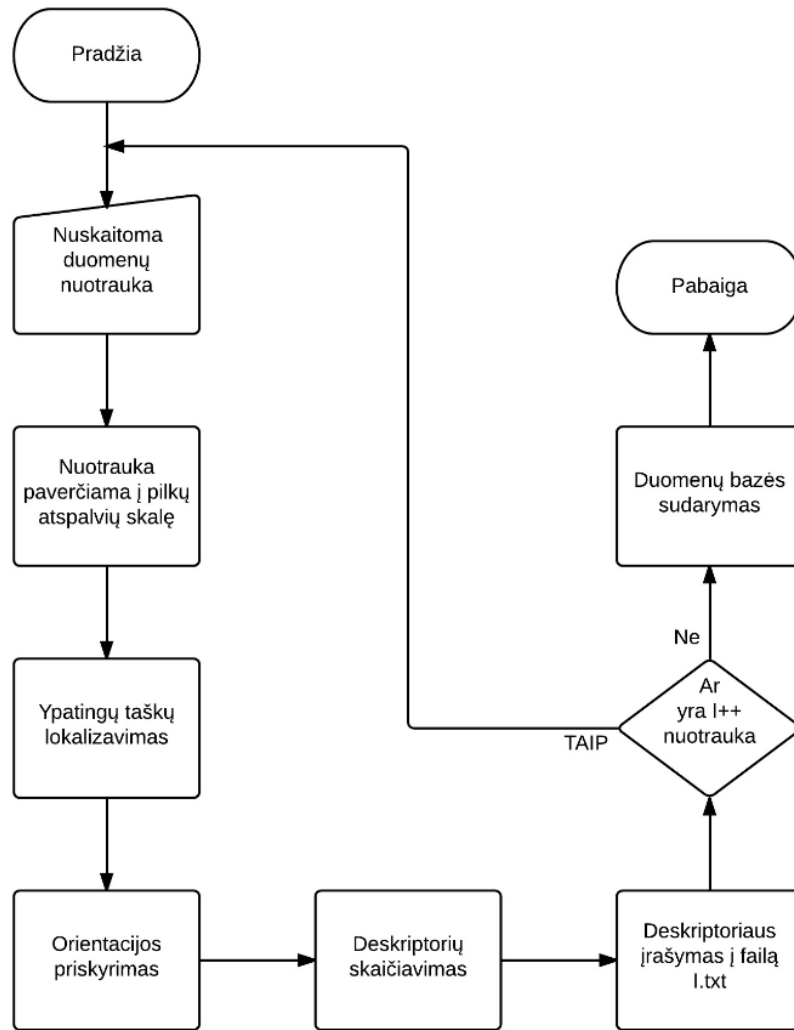
13. Pav. Antroji duomenų bazė 90° kampu.

Tada reikia atlikti nuotraukos mastelio mažinimą. Gautos nuotraukos iš fotoaparato kameros dydis yra 4320 x 3240 pikselių. Atlikus nuotraukų mastelio mažinimą, nuotraukų dydis yra sumažinamas iki 800 x 600 pikselių. Keičiant nuotraukos dydį svarbu išlaikyti santykį tarp nuotraukos kraštinių 4:3, priešingu atveju nuotrauka gausis iškraipyta, ko pasekoje gausis iškraipyti nuotraukos ypatingieji taškai, taip suprastinant vaizdo atpažinimo tikslumą.

Siekiant įvertinti, kuri duomenų bazė yra geresnė ir tinkamesnė praktiniame naudojime, visi bandymai bus atliekami su abiem duomenų bazėmis, kurių metu bus nustatyta duomenų bazės universalumas bei naudotinių metodų tikslumas, ir bus nustatyta, kuri duomenų bazė yra geresnė.

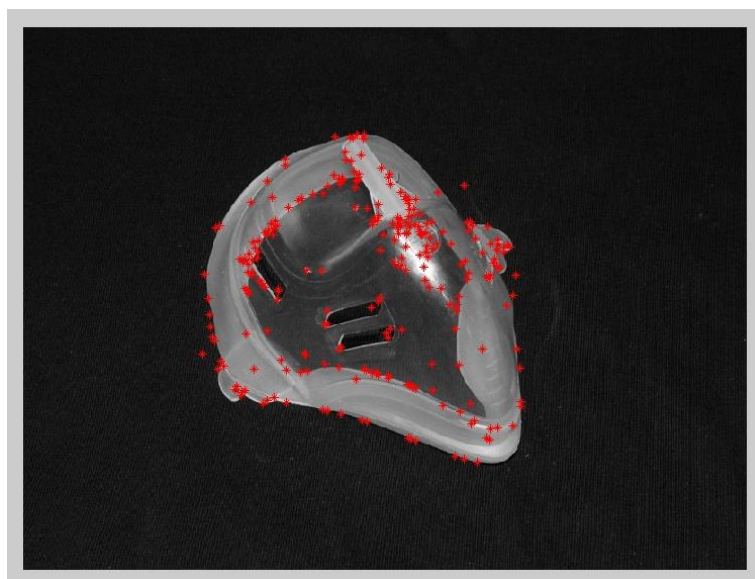
2.1.1 Duomenų bazės sudarymo algoritmas SIFT metodu.

Kai nuotraukos yra paruoštos naudoti, tada iš jų reikia išgauti konkrečios nuotraukos ypatinguosius taškus, kurie kiekvienoje nuotraukoje yra unikalūs. Žemiau pateiktame 14 paveiksle, yra pavaizduotas algoritmas, kuris iš nuotraukų SIFT metodu išgauna ypatingųjų taškų deskriptorius.



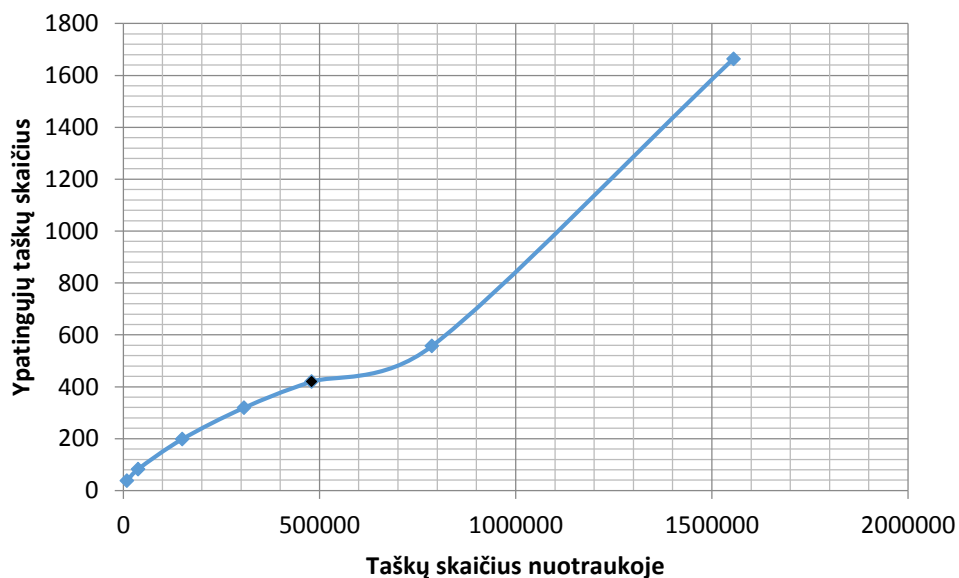
14. Pav. SIFT metodu pagrįstas duomenų bazės sudarymas.

Pradžioje algoritmo, yra nuskaitoma pirmoji duomenų bazės nuotrauka („ $I = 1$ “), kuri yra konvertuojama į pilkų atspalvių nuotrauką (MATLAB programoje tai atlieka funkcija: *rgb2gray*). Tada yra nustatomi galimai tai nuotraukai ypatingieji taškai – vyksta ypatingųjų taškų lokalizacija. Po to yra tikrinama, ar šie ypatingieji taškai gali būti priskirti ypatingiesiems taškams, ir vyksta jų orientacijos priskyrimas. Gavus ypatingųjų taškų (žr. 15 paveikslą) deskriptorius, jie yra įrašomi į failą (šiuo atveju „l.txt“), ir tikrinama ar dar yra nuotraukų duomenų bazėje. Jei egzistuoja sekanti nuotrauka duomenų bazėje, tai algoritmas vėl kartojamas iš naujo, tik šiuo atveju $I = 2$. Kai duomenų bazėje nebėra naujų nuotraukų yra sudaroma duomenų bazė ir ciklas yra baigiamas.



15. Pav. SIFT algoritmo pagrindu rasti ypatingieji taškai.

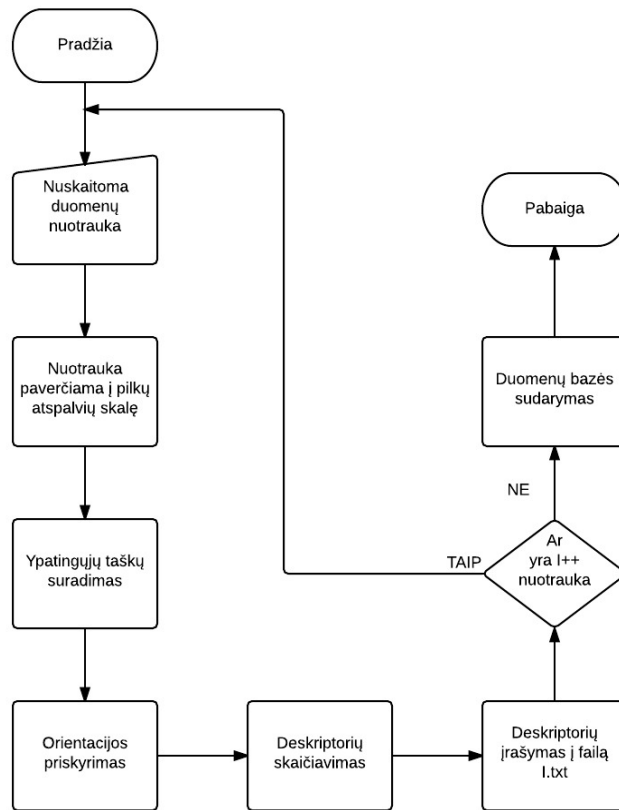
Žemiau pateiktame grafike, yra pavaizduota SIFT ypatingųjų taškų kiekis nuotraukoje priklausomai nuo nuotraukos dydžio. Kaip matyti, kuo mažesnė nuotrauka yra duomenų bazėje, tuo surandama mažiau ypatingųjų taškų, ko pasekoje greitėja algoritmo veikimas. Šiuo atveju, kai nuotraukos dydis yra 800x600 ypatingieji taškai iš nuotraukos yra išgaunami per ~3-10s.



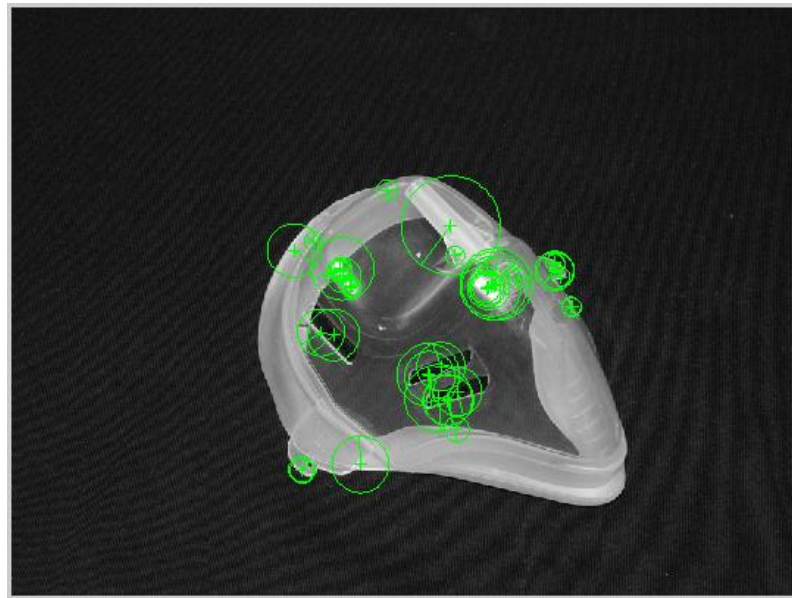
16. Pav. SIFT ypatingųjų taškų skaičiaus priklausomybė nuo nuotraukos dydžio.

2.1.2 Duomenų bazės sudarymo algoritmas SURF metodu.

17 paveiksle, yra pavaizduotas algoritmas, kuris iš duomenų bazės nuotraukų, SURF metodu, išgauna ypatingųjų taškų deskriptorius (žr. 18 paveikslą).

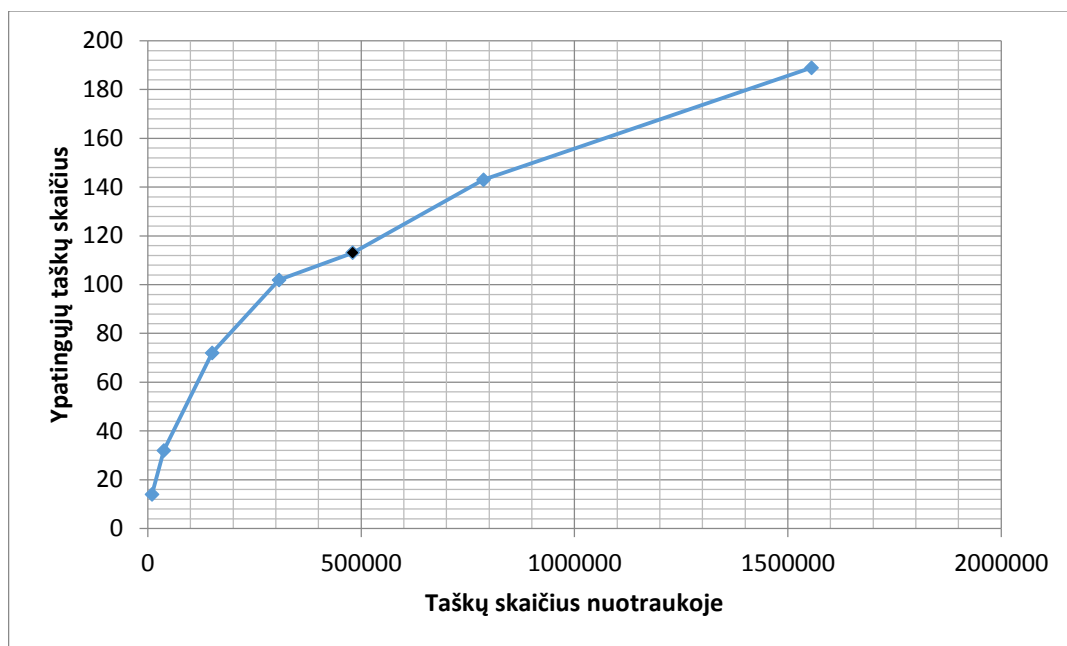


17. Pav. SURF metodu pagrįstas duomenų bazės sudarymas.



18. Pav. SURF algoritmo pagrindu rasti ypatingieji taškai

Žemiau pateiktame grafike matyti SURF metodu gautų ypatingųjų taškų skaičius priklausomai nuo nuotraukos dydžio. Iš čia matosi, kad SURF metodu yra surandama, kai nuotraukos dydis 800x600, apie tris kartus mažiau ypatingųjų taškų nei naudojant SIFT metodą. SURF metodu iš vienos nuotraukos ypatingieji taškai gaunami per ~1-5s.

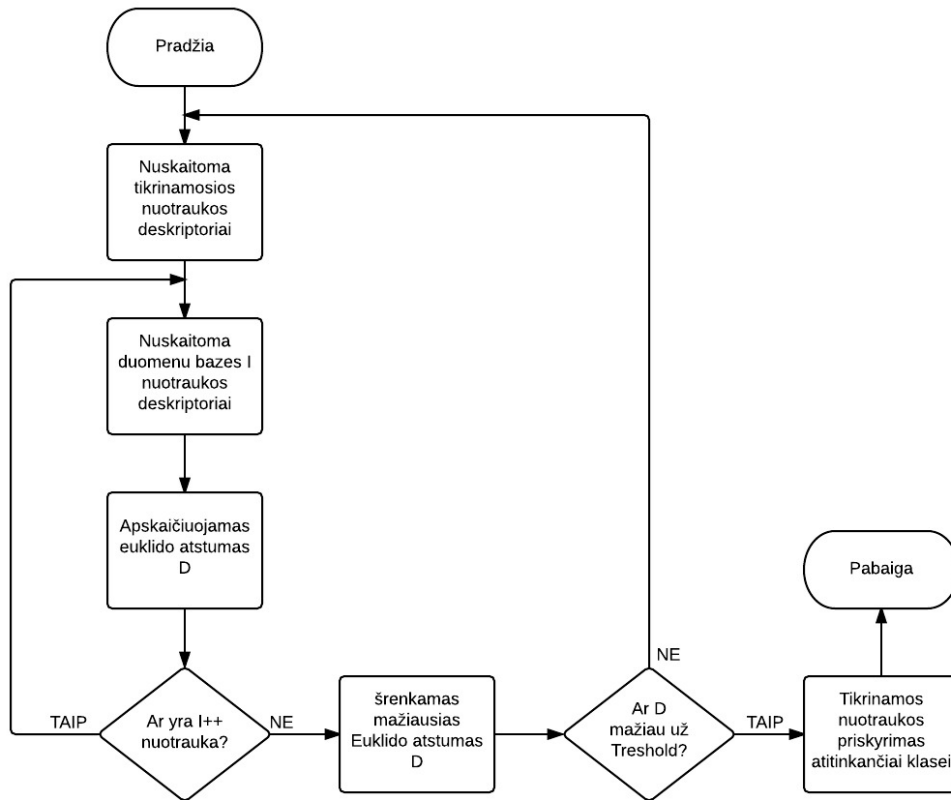


19. Pav. SURF ypatingųjų taškų skaičiaus priklausomybė nuo nuotraukos dydžio.

2.2 Euklido atstumu pagrįstas kvėpavimo kaukės atpažinimo algoritmas

Sudaryta duomenų bazė yra patikrinama su testavimo nuotraukomis, kurios nebuvo naudotos sudarant duomenų bazę. Euklido atstumu pagrįstas kvėpavimo kaukės atpažinimo algoritmas yra parodytas 20 paveiksle, o šio algoritmo realizavimas MATLAB programavimo aplinkoje parodytas 21 paveiksle.

Algoritmas yra pradedamas, nuskaitant pirmosios testuojamos nuotraukos deskriptorius. Vėliau yra nuskaitoma pirmosios nuotraukos deskriptoriai iš duomenų bazės. Tada yra apskaičiuojami kiekvienos testuojamos nuotraukos t. y, kiekvieno deskriptoriaus Euklido atstumas tarp tiriamos duomenų bazės kiekvieno deskriptoriaus, ir randamas mažiausią atstumą turintys deskriptoriai. Kuo mažesnis Euklido atstumas yra, tuo tikrinami vektoriai yra panašesni.



20. Pav. Klasifikavimo algoritmas pagrįstas Euklido atstumu.

```

threshold = 30;
distRatio = 0.7;

for j = 1:1:23
    for k = 1:1:308
        for i = 1 : size(des1,1)
            dotprods = des1(i,:) * des2;
            [vals,indx] = sort(acos(dotprods));
            if (vals(1) < distRatio * vals(2))
                match(i) = indx(1);
            else
                match(i) = 0;
            end
        end
        num(k) = sum(match > 0);
        if (num(k) > threshold)
            ClassNr(j,k) = index(k);
        else
            ClassNr(j,k) = 0;
        end
    end
end
end
  
```

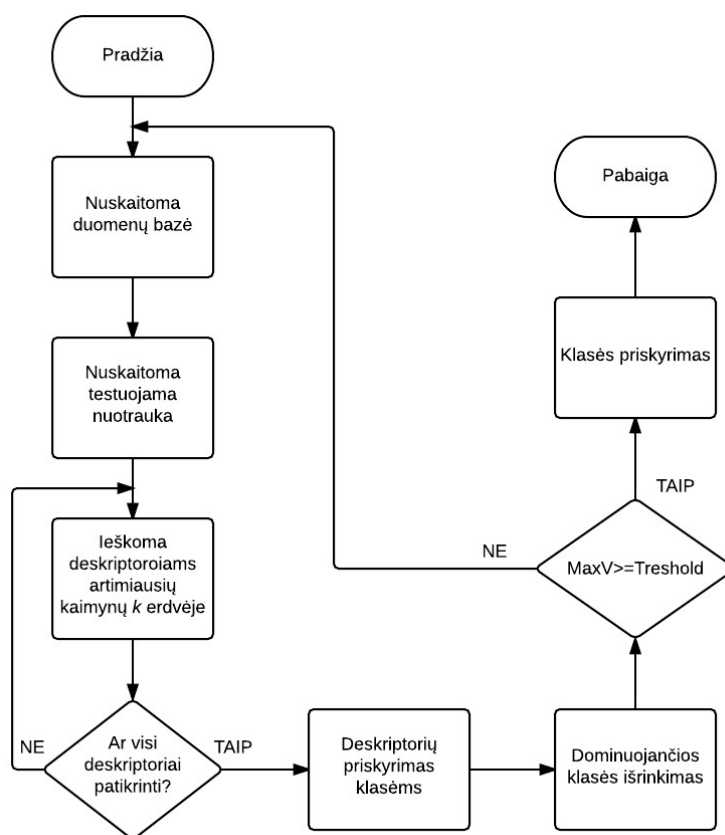
21. Pav. Euklido atstumu klasifikuojamų kaukių algoritmas MATLAB programoje.

Kadangi Euklido atstumu gauta metrika yra asimetrinė, tai nereiškia, kad atrinktas artimiausias deskriptorius iš duomenų bazės bus artimiausias atlikus atvirkštinį veiksmą. Todėl yra įvedamas į algoritmą (žr. 21 paveikslą) reikšmingas slenkstis (angl. distRatio). Tai reiškia, kad jei Euklido atstumas iki artimiausio kaimyno yra mažesnis, nei nustatytas slenkstis, tai šis deskriptorius yra priskiriamas tai klasei, kuriai priklausė deskriptorius, priešingu atveju, tas deskriptorius yra atmetamas.

Tada yra tikrinama kiek tikrinamosios nuotraukos deskriptorių iš duomenų bazės atitiko atsitiktiniai testuojamai nuotraukai, ir jei ta suma yra didesnė, už nustatytą ribinį slenkstį (angl. Threshold), testuojama nuotrauka yra priskiriama tai klasei, kuriai priklausė lyginama kaukė iš duomenų bazės.

Algoritmas veikia tol, kol testuojama nuotrauka patikrinama su visa duomenų baze, nes ši nuotrauka gali būti priskiriama nebūtinai tik vienai klasei.

2.3 K-NN artimiausių kaimynų klasifikavimo algoritmas



22. Pav. Klasifikavimo algoritmas pagrįstas artimiausio kaimyno metodu.

21 paveiksle yra pavaizduotas K-NN artimiausių kaimynų algoritmo veikimo principas. Pradžioje, yra nuskaitoma duomenų bazė ir ji suskirstoma į atitinkamas klases, pagal kvėpavimo kaukę. Tada yra tikrinamas testuojamos kaukės kiekvienas deskriptorius su duomenų bazės deskriptoriais, ir kiekvienam deskriptoriui yra priskiriama pagal taisyklės artimiausia klasė. Kai deskriptoriai yra priskirti klasėms, yra išrenkama dominuojanti klasė, ir jei tai klasei priklausančių deskriptorių suma yra didesnė, už nustatytą ribinį slenkstį (angl. $MaxV \geq Treshold$), tai ta klasė yra priskiriama tai nuotraukai, priešingu atveju, klasė yra nepriskiriama.

MATLAB programoje K-NN funkcija yra aprašoma taip:

```
„class = knnclassify(sample, training, group, k, 'distance', 'rule')“,
```

Kur *sample* – yra tikrinamosios nuotraukos deskriptoriai, *training* – duomenų bazė, *group* – klasių priskyrimas deskriptoriams, *k* artimiausių kaimynų skaičius tikrinamoje erdvėje, „*distance*“ (atstumas) – taisyklė, kurioje nurodoma, koku būdu bus skaičiuojamas atstumas tarp artimiausių kaimynų, o „*rule*“ (taisyklė) taisykle nurodoma kaip nuspręsti klasės priskyrimą. Šiame magistriniame darbe „*distance*“ atstumui nustatyti naudotos šios taisyklės:

- Euklido atstumas;
- „*cityblock*“ – absoliučių skirtumų suma;
- „*cosine*“ – kosinuso kampas tarp tiriamų
- „*correlation*“ – vienas minus koreliacijos atstumas tarp taškų.

Naudojamos taisyklė (angl. *rule*) klasių priskyrimui:

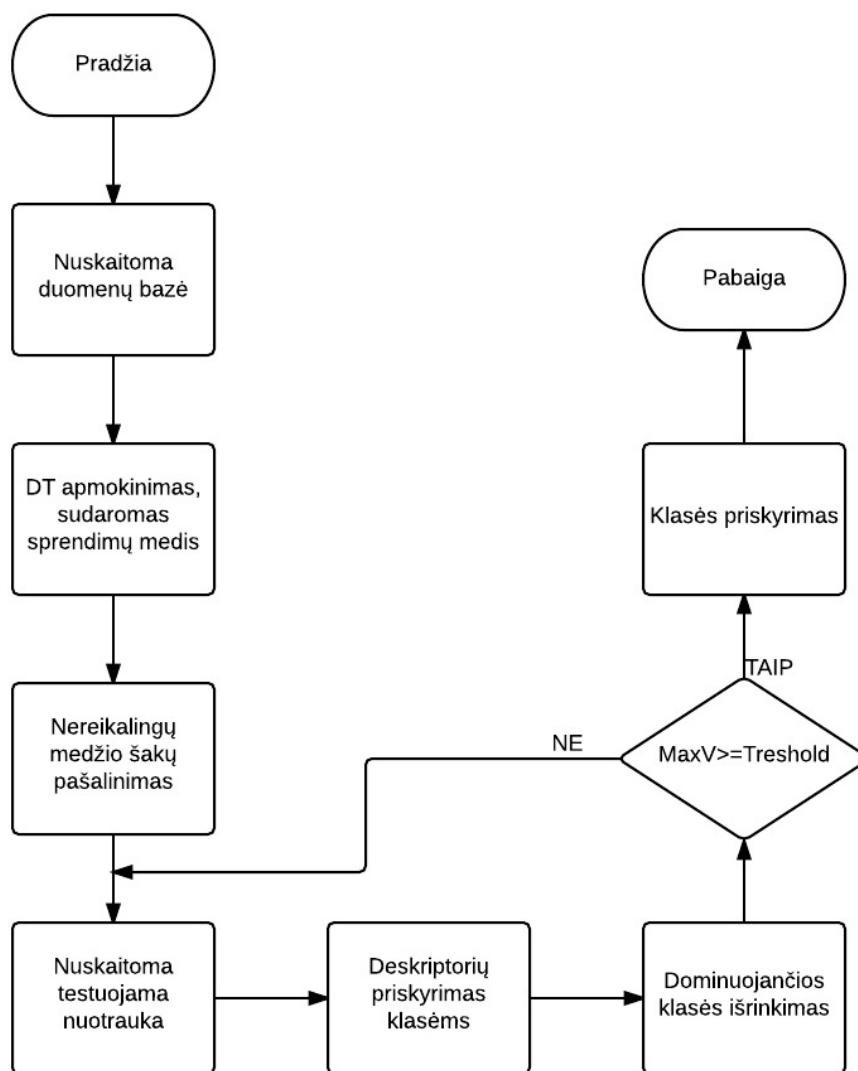
- Artimiausias (angl. *nearest*) – artimiausios klasės priskyrimas tikrinamojoje erdvėje;
- Atsitiktinė (angl. *random*) – atsitiktinio deskriptoriaus priskyrimas tikrinamojoje erdvėje;
- *Consensus* taisyklė klasių priskyrimui – kai *k* erdvėje yra kelių klasių deskriptoriai, tai šiam deskriptoriui nėra priskiriama klasė (priskiriama MATLAB programoje *NaN* (jokia) reikšmė).

2.4 DT sprendimo medžio klasifikavimo algoritmas

Kaip matyti žemiau pateiktame paveiksle (žr. 23 paveikslą), algoritmas pradeda veikti nuskaitydamas duomenų bazės deskriptorius, su iš karto kiekvienam deskriptoriui priskirta atitinkama klasė. Tada vyksta algoritmo apmokymas, kurio metu yra sukuriamas medis, su šakomis, kur kiekviena šaka atitinka tam tikrą taisyklę, bei lapais, kur kiekvienas lapas atitinka tam tikrą klasę.

Kai algoritmas apsimoko, t. y. sukuriamas medis, su šakomis bei lapais, vyksta medžio „gedėjimas“ (angl. *pruning*), kurio metu yra apkarpomis nereikalingos šakos, kurios tik stabdo

algoritmo veikimą, bet įtakos jo veikimui neturi. Tada yra nuskaitymi testuojamos nuotraukos ypatingieji taškai, kurie pradami tikrinti, nuo medžio šaknų, ir pritaikius taisykles, kiekvienas deskriptorius yra nukreipiamas žemyn į lapus ir pagal lapo klasę yra priskiriama deskriptoriui atitinkama klasė. Kai deskriptoriums klasės jau yra priskirtos, išrenkamos atitinkamai nuotraukai dominuojanti klasė. Jei dominuojančios klasės deskriptorių suma yra didesnė už nustatytą ribinį slenkstį (angl. $MaxV \geq Treshol$), tai ši klasė priskiriama tikrinamai nuotraukai, priešingu atveju nuotrauka atmetama ir nepriskiriama nei vienai klasei.



23. Pav. DT – sprendimo medžio klasifikatoriaus veikimo algoritmas.

Medžio mokymas MATLAB programinėje aplinkoje atitinka šiai funkcijai:

```
tree1 = fitctree(total,ccc, 'Prior', 'uniform', 'MinLeaf', X, 'MinParent', Y);
```


čia komanda „*prior*, ‘*uniform*‘“ liepiama medžiui kiekvienai klasei priskirti po lygiai šakų ir lapų, nes priešingu atveju gali atsitikti taip, kad kuri nors klasė taps dominuojanti, ko pasekoje krenta algoritmo tikslumas. Komanda „*MinLeaf*, ‘*X*‘“ yra nurodomas minimalus lapų skaičius, kuris turi būti apmokytame medyje, o komanda „*MinParent*, ‘*Y*‘“ yra apibrėžiamas minimalus šakų skaičius. Šiuo atveju, minimalus šakų skaičius negali būti mažesnis už lapų skaičių.

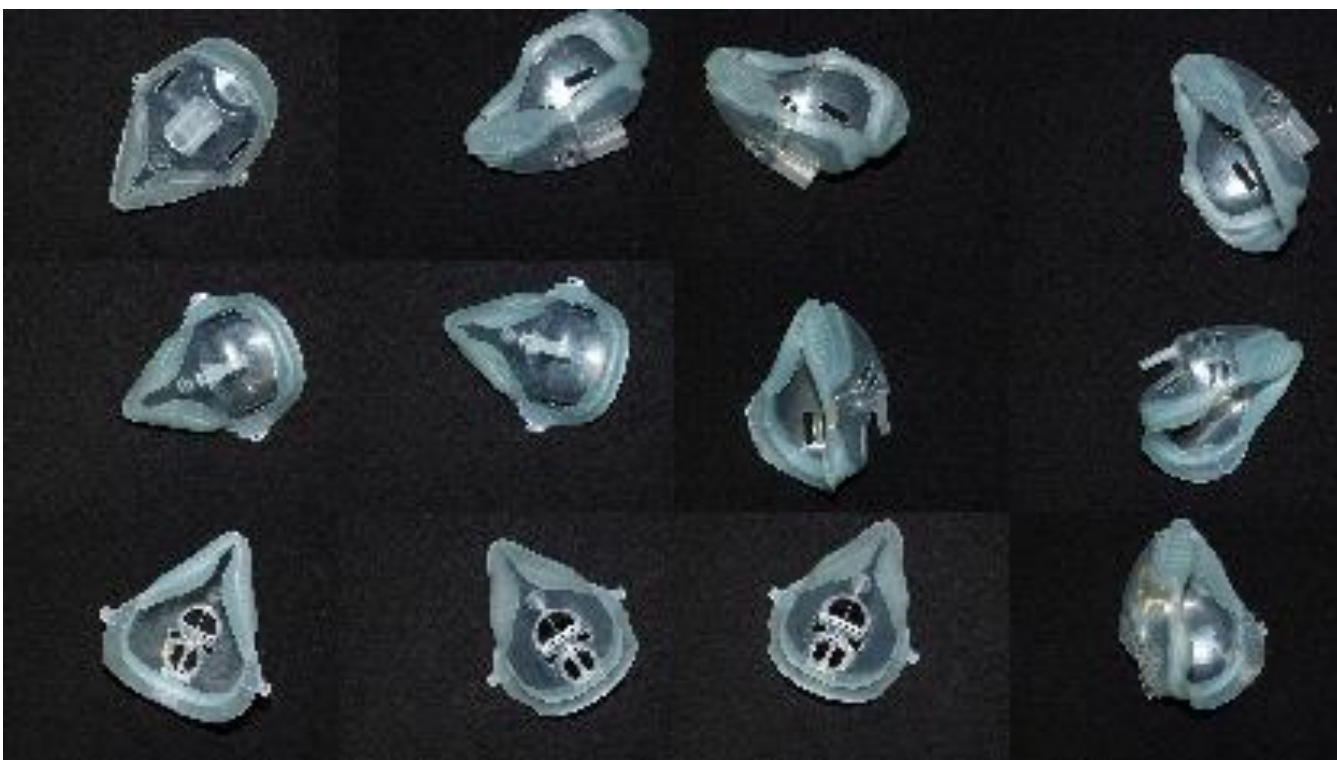
Sprendimo medžio algoritmas nuo artimiausių kaimynų algoritmo skiriasi tuo, kad šiame algoritme pirmiausia vykdomas sprendimo medžio apmokymas, o tik vėliau yra tikrinamos testuojamos nuotraukos.

3 KVĖPAVIMO KAUKIŲ KLASIFIKAVIMO ALGORITMO EKSPERIMENTINIAI TYRIMAI

Eksperimentinėje dalyje yra tiriama kvėpavimo kaukių klasifikavimo algoritmai. Tam, kad visapusiškai ištirti šių algoritmų veikimą, buvo suformuotos bandomosios duomenų bazės, po 12 nuotraukų kiekvienai duomenų bazei ištirti. Testavimo nuotraukų imtyje yra po 4 kiekvienos klasės nuotraukas, kurios nebuvo naudojamos mokymuose (žr. 24, 25 paveikslą. Kiekviena testavimo imtis yra sudaryta iš nuotraukų, užfiksuotų tokiomis pačiomis sąlygomis kaip ir duomenų bazė, esant tam pačiam fotografavimo kampui bei apšvietumui.



24. Pav. Testuojamos nuotraukos su A duomenų baze.



25. Pav. Testuojamos nuotraukos su B duomenų baze.

Eksperimentinių tyrimų metu bus atsižvelgta į įvairius algoritmo modifikacijas, skirtingus ypatingųjų taškų gavimo algoritmus bei skirtingus klasifikavimo algoritmus:

- SIFT + Euklido atstumas;
- SURF + Euklido atstumas;
- SIFT + K-NN artimiausių kaimynų algoritmas;
- SURF + K-NN artimiausių kaimynų algoritmas;
- SIFT + DT sprendimų medžio algoritmas;
- SURF + DT sprendimų medžio algoritmas.

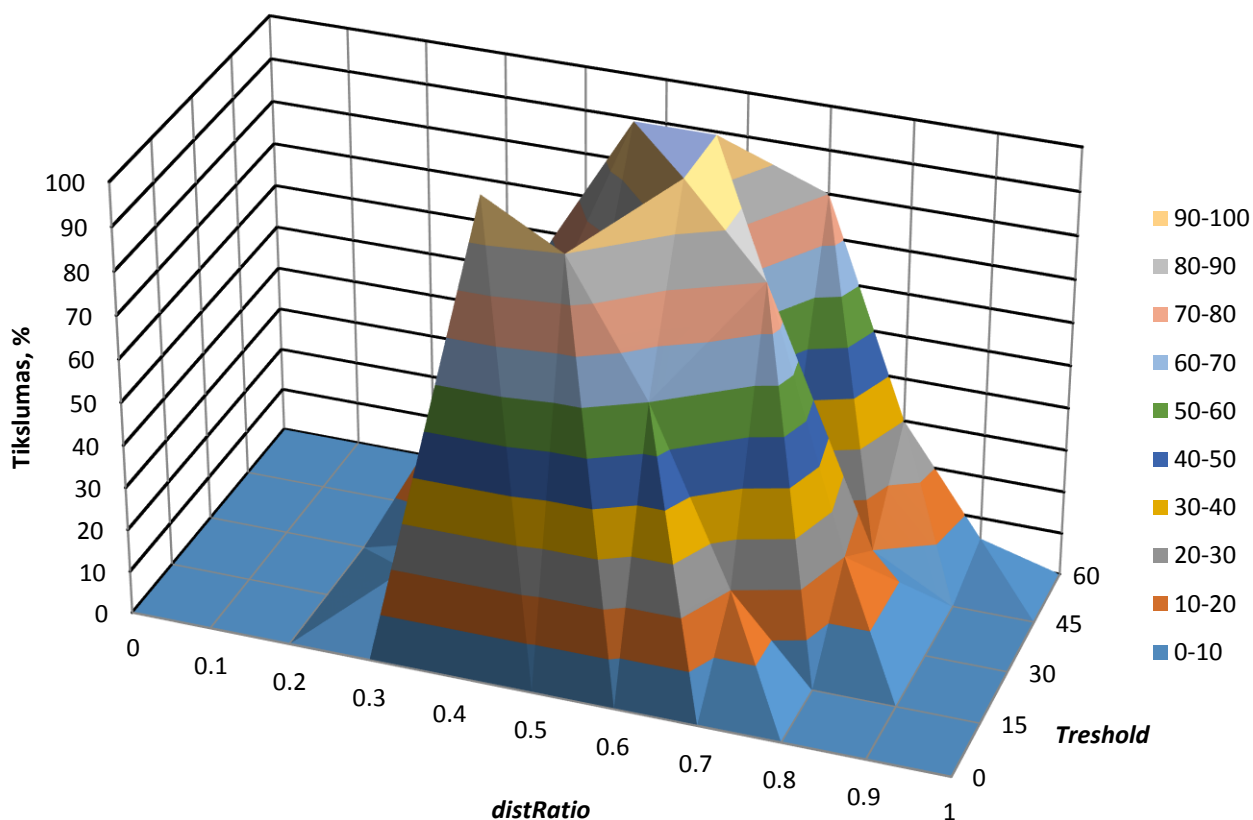
Eksperimentinių tyrimų metu algoritmai bus tiriami, atsižvelgiant į šiuos parametrus:

- Nuo parametro *distRatio* reikšmės;
- Nuo parametro *Threshold* reikšmės;
- Skirtingos duomenų bazės įtaka atpažinimo kokybei;
- Skirtingų algoritmų parametrų įtaka;
- Algoritmo modifikacijų greitimeika;
- Algoritmo modifikacijų atpažinimo tikslumas.

3.1 Euklido atstumu pagrįsto klasifikavimo algoritmo eksperimentiniai tyrimai

Šiame skyriuje bus parodyti eksperimentiniai tyrimai, kuriais bus tikrinamas algoritmo atpažinimo tikslumas nuo skirtingos *distRatio* bei *Threshold* reikšmės. Pirmas bandymas buvo atliekamas su tokiomis reikšmėmis: *distRatio* = 1 *Threshold* = 100. Su šiomis parametru vertėmis algoritmas neišskyrė jokių kvėpavimo kaukių. Su šiais duomenimis atpažinimo kokybė lygi 0%, todėl įvertinus tai, eksperimentiniai tyrimai bus atliekami su tokiomis *distRatio* reikšmėmis: 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1, *Threshold* reikšmės: 15, 30, 45, 60.

Euklido atstumu pagrįstas klasifikavimo algoritmas naudojant SIFT metodą ir A duomenų bazę



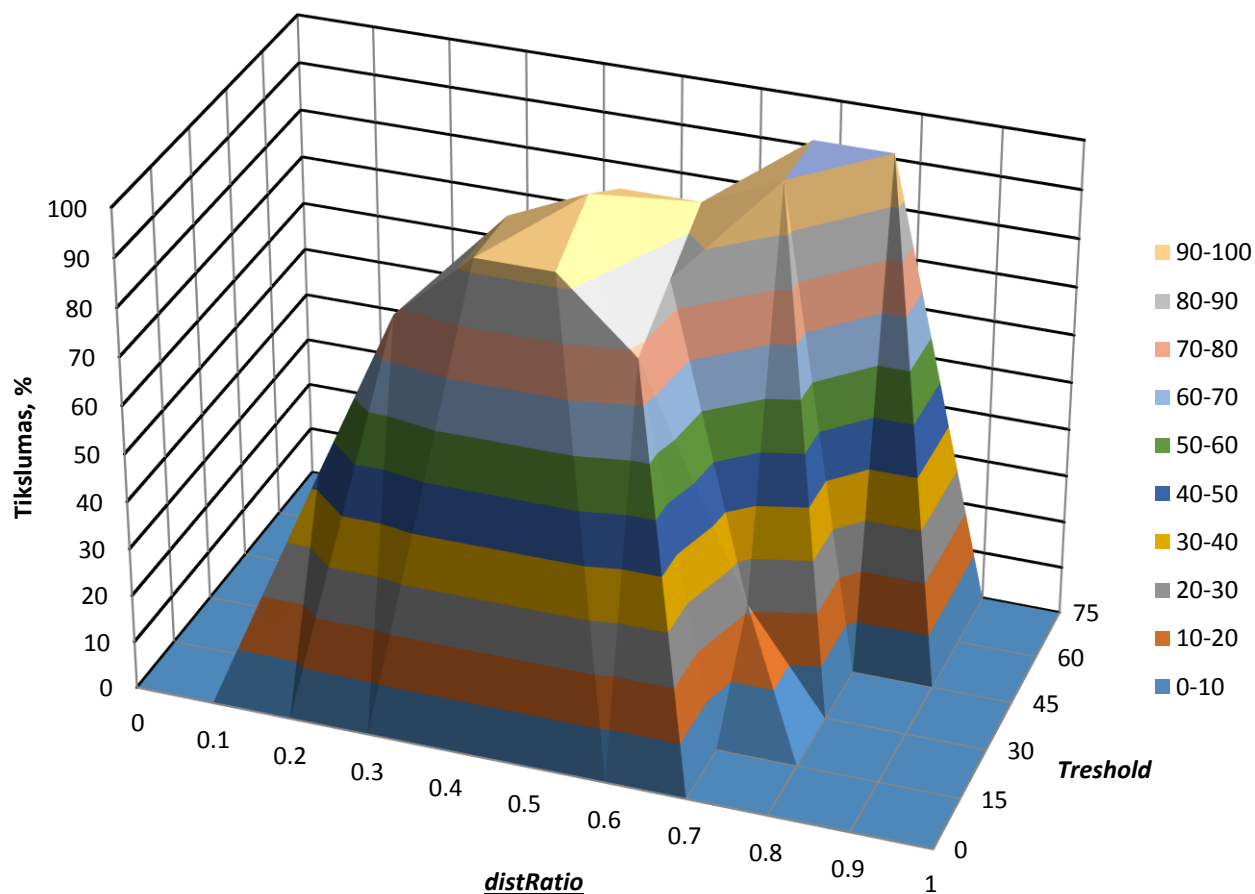
26. Pav. Klasifikavimo algoritmo, naudojant Euklido klasifikatorių su SIFT ypatingųjų požymių išskyrimo metodu ir A duomenų baze, tikslumas.

Aukščiau pateiktame tikslumo paviršiuje yra pavaizduoti tyrimo rezultatai naudojant Euklido atstumu pagrįstą klasifikavimo algoritmą su A duomenų baze, kur ypatingieji taškai yra gauti naudojant SIFT metodą.

Iš tikslumo paviršiaus matyti, kad geriausi rezultatai buvo pasiekti, kai *distRatio* reikšmės kito nuo 0.3 iki 0.7 ir siekė 100%. Naudojant šį metodą, 100% tikslumas buvo pasiektas su šiomis parametru reikšmėmis: *Threshold*: 15 ir *distRatio*: 0.3, *Threshold*: 30 ir *distRatio*: 0.6, bei *Threshold*: 45 ir *distRatio*: 0.5, 0.6.

Žemiau yra pateiktas eksperimentinis tyrimais naudojant Euklido atstumu pagrįsta klasifikatorių, su SIFT metodu gautais deskriptoriais, bei B duomenų baze.

Euklido atstumu pagrįstas klasifikavimo algoritmas naudojant SIFT metodą ir B duomenų bazę

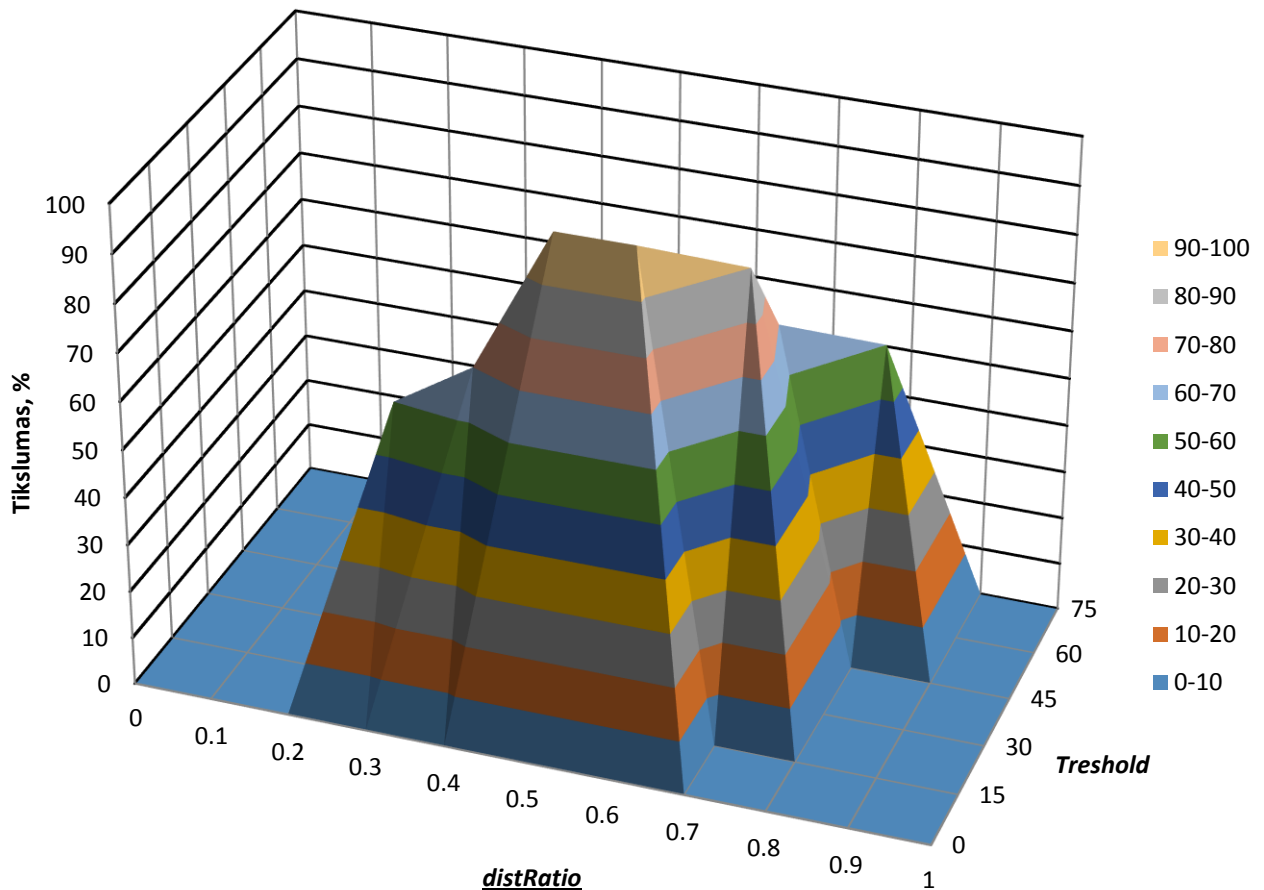


27. Pav. Klasifikavimo algoritmo, naudojant Euklido klasifikatorių su SIFT ypatingųjų požymių išskyrimo metodu ir B duomenų baze, tikslumas.

Kaip matyti iš gauto paviršiaus, naudojant šią duomenų bazę geriausi rezultatai buvo pasiekti, kai *distRatio* reikšmės yra: 0.5, 0.7, 0.8 ir buvo pasiektas 100% tikslumas. Esant šioms parametru reikšmėms, buvo pasiekti geriausi rezultatai: *Threshold*: ir *distRatio*: 0.5, *Threshold*: 45 ir *distRatio*: 0.7, bei *Threshold*: 60 *distRatio*: 0.7, 0.8.

Sekantys tyrimai buvo atliekami su tuo pačiu Euklido atstumo klasifikavimo algoritmu, tik šį kartą duomenų bazė buvo sudaryta iš SURF metodu gautų ypatingųjų taškų deskriptorių. Tyrimo rezultatai yra pavaizduoti 28 paveiksle su A duomenų baze, o 29 paveiksle yra parodyti rezultatai su B duomenų baze.

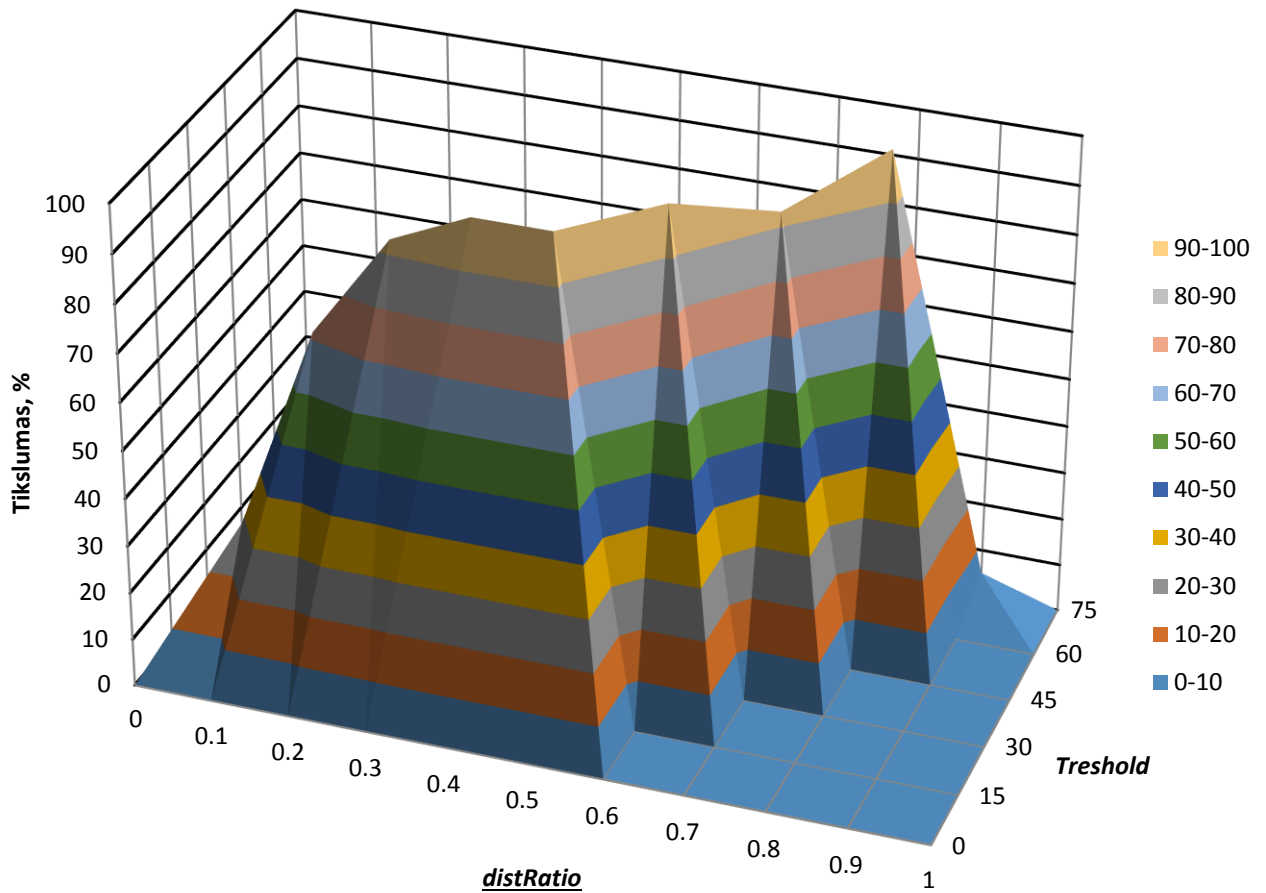
Euklido atstumu pagrįstas klasifikavimo algoritmas naudojant SURF metodą ir A duomenų bazę



28. Pav. Klasifikavimo algoritmo, naudojant Euklido klasifikatorių su SURF ypatingųjų požymių išskyrimo metodu ir A duomenų baze, tikslumas

Kaip matyti iš gauto paviršiaus aukščiau, naudojant SURF metodu gautus ypatingųjų taškų deskriptorius taip pat yra pasiekiamas 100% kvėpavimo kaukių priskyrimas tam tikrai klasei. Tik šį kartą tai yra pasiekama, su šiomis parametru reikšmėmis: *Threshold*: 15 ir *distRatio*: 0.5, 0.6.

Euklido atstumu pagrįstas klasifikavimo algoritmas naudojant SURF metodą ir B duomenų bazę



29. Pav. Klasifikavimo algoritmo, naudojant Euklido klasifikatorių su SURF ypatingųjų požymių išskyrimo metodu ir B duomenų baze, tikslumas.

Aukščiau pavaizduotame paviršiuje yra parodyti tyrimo rezultatai, naudojant Euklido klasifikatorių, kur ypatingųjų taškų deskriptoriai yra gauti naudojant SURF metodu iš B duomenų bazės. Kaip matyti iš paviršiaus, tyrimo rezultatai parodė kad tikslumas kinta gan plačiose ribose, kai *distRatio* kinta nuo 0.2 iki 0.8 priklausomai nuo *Threshold*. Geriausias rezultatas yra 100% tikslumas ir jis pasiektas su šiais algoritmo parametrais: *Threshold*: 15 ir *distRatio*: 0.4, 0.5, *Threshold*: 30 ir *distRatio*: 0.6., *Threshold*: 60 ir *distRatio*: 0.8.

Kaip matyti iš tyrimų rezultatų, naudojant šį metodą su SIFT ir SURF gautais ypatingaisiais taškais, buvo pasiekti labai geri rezultatai, kurie siekė 100%. Kaip matyti iš paviršių, geriausi rezultatai pasiekiami gan siaurame parametrų diapazone. Vieninteliu atveju, kai buvo naudojamas SIFT metodu gauti ypatingieji taškai iš B duomenų bazės, geri rezultatai pasiekiami didesniame diapazone, nes SIFT metodu gauti ypatingųjų taškų deskriptoriai yra sudaryti iš vektoriaus, kurio ilgis

yra 128 skaitmenys, o SURF metodu gauti ypatingųjų taškų vektorių ilgis yra tik 64. Tačiau naudojant SURF metodą buvo pasiekta daug didesnė greitaveika nei naudojant SIFT metodą. 2 lentelėje yra parodyti tyrimo metu gautas vidutinis vienos nuotraukos priskyrimas atitinkamai klasei laikas:

2. lentelė. Klasifikavimo algoritmo, pagrįsto Euklido atstumu, greitaveika.

Ypatingųjų taškų gavimo metodas	SIFT		SURF	
	A	B	A	B
Duomenų bazė				
Vidutinis vienos nuotraukos patikrinimo laikas, s	60.2	50.3	30.3	28.7

Kaip matyti 2 lentelėje, naudojant SURF metodą, nuotrauka yra patikrinama dvigubai greičiau nei naudojant SIFT metodą, o tai atsitinka dėl to, kad ypatingieji taškai yra aprašomi dvigubai trumpesniu deskriptoriumi, ir šis metodas kiekvienoje nuotraukoje randa apie 3 kartus mažiau ypatingų taškų, ko pasekoje ir gaunamas daug trumpesnis tikrinimo laikas.

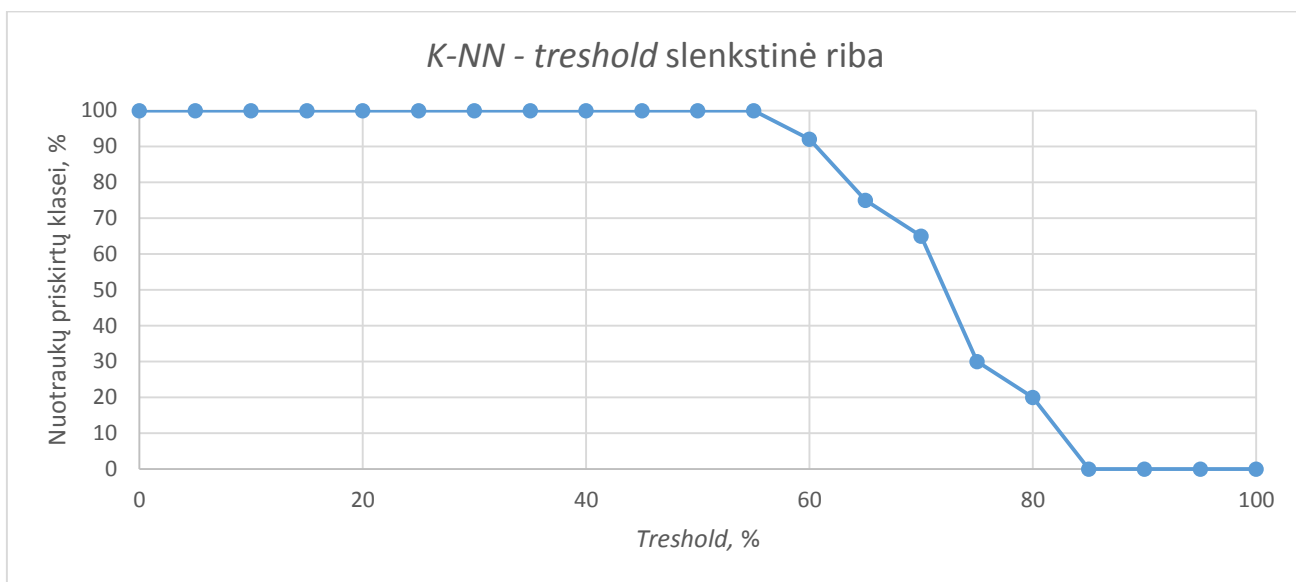
3.2 K-NN artimiausių kaimynų metodu pagrįstas klasifikavimo algoritmo tyrimai

Šiame skyriuje bus parodyti kvėpavimo kaukių atpažinimo tikslumo tyrimai, naudojant K-NN klasifikavimo algoritmą, bei įvairias jo modifikacijas. Visi tyrimai yra atliekami su trimis k (artimiausių kaimynų skaičius) reikšmėmis: 1, 2, 5. Taip pat visi tyrimai yra atliekami su keturiais atstumo nustatymo nuo tiriamo deskriptoriaus metodais: *Euklido atstumas*; „cityblock“; „cosine“; „correlation“. Bei nuo skirtingų taisyklių klasių priskyrimui: *Artimiausias* (angl. *nearest*); *Atsitiktinė* (angl. *random*); *Consensus taisyklė*.

Įvertinus tai, kad kiekvienos tikrinamos nuotraukos deskriptorių šis algoritmas turi priskirti kuriai nors klasei, pradžioje buvo atliekamas *Threshold* ribinio slenksčio nustatymo tyrimas. Tyrimas buvo atliekamas siekiant nustatyti, kokią dalį deskriptorių iš visų turi priskirti vienai klasei šis algoritmas, kad nuotrauka būtų priskirta atitinkamai klasei. Įvertinus tai, kad kiekvienoje tikrinamoje nuotraukoje deskriptorių skaičius yra skirtingas, reikia nustatyti slenkstinę ribą, kurią peržengus algoritmas priskirtų tiriamą nuotrauką atitinkamai klasei. Šio tyrimo rezultatai yra pavaizduoti 30 paveiksle.

Kaip matyti iš grafiko, iki *threshold* 55% ribos, algoritmas 100% visas nuotraukas priskiria atitinkamai klasei, o nuo 55% priskyrimo procentas pradeda kristi, ir ne visos tiriamos nuotraukos yra priskiriamos atitinkamai klasei. Todėl ribinio slenksčio *Threshold* riba šiam algoritmui yra parenkama

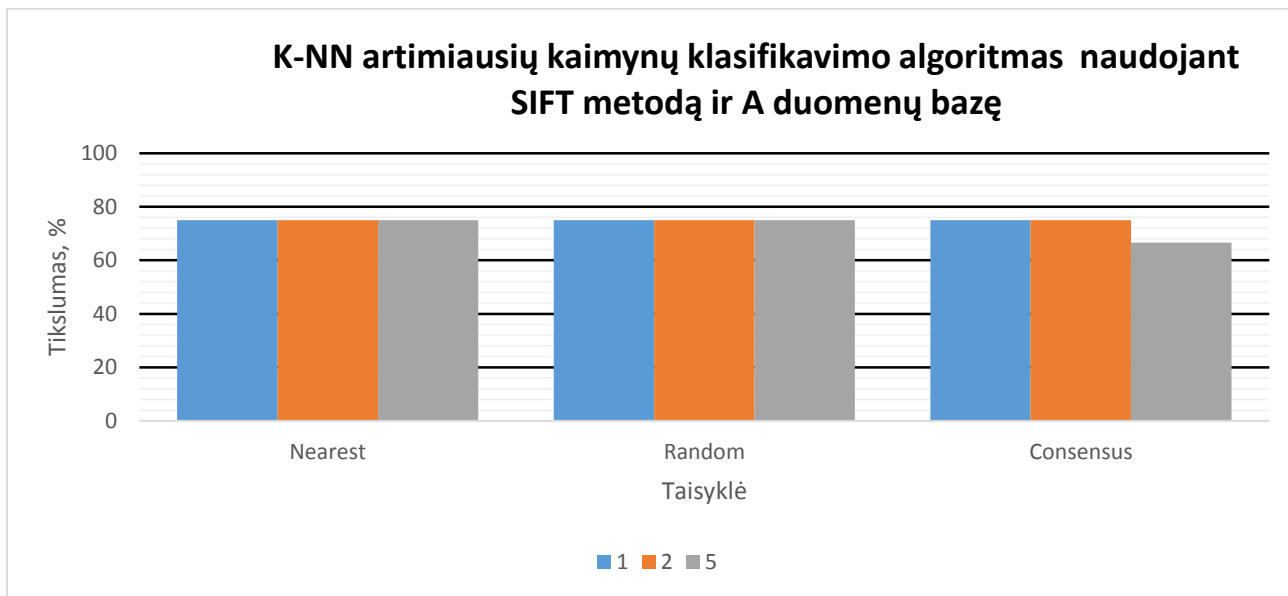
55% t. y. jei algoritmas 55% visų deskriptorių priskirs kuriai nors vienai klasei, tai ta klasė bus priskiriama tikrinamai nuotraukai.



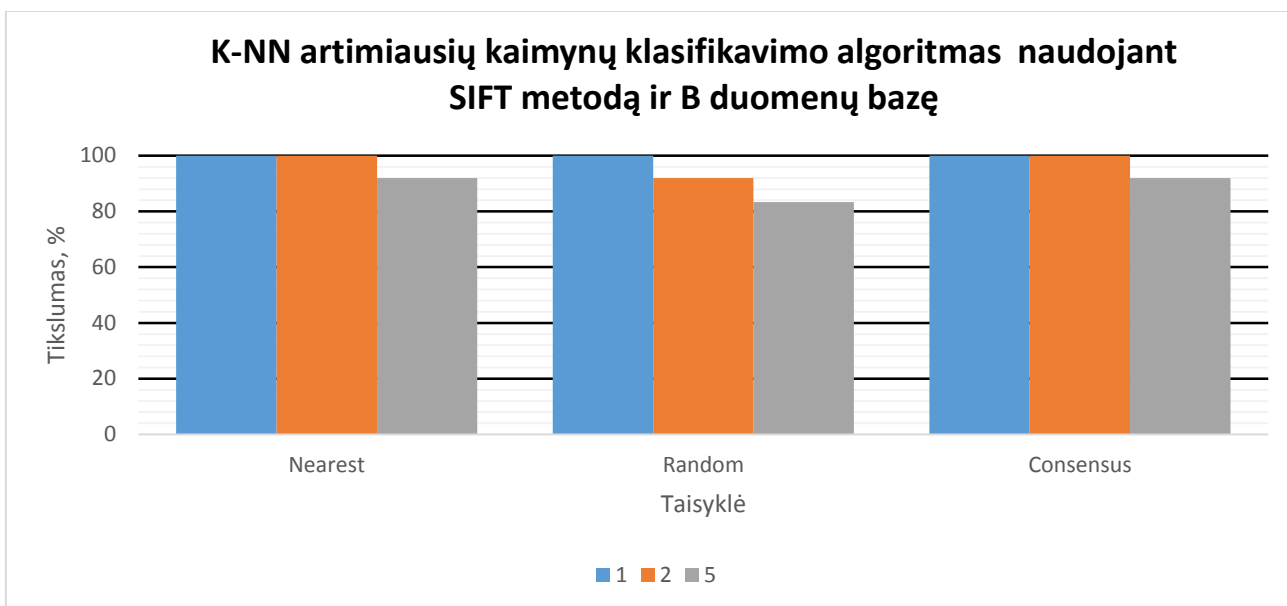
30. Pav. Parametro *Threshold* nustatymas.

3.2.1 Euklido atstumo taisyklė

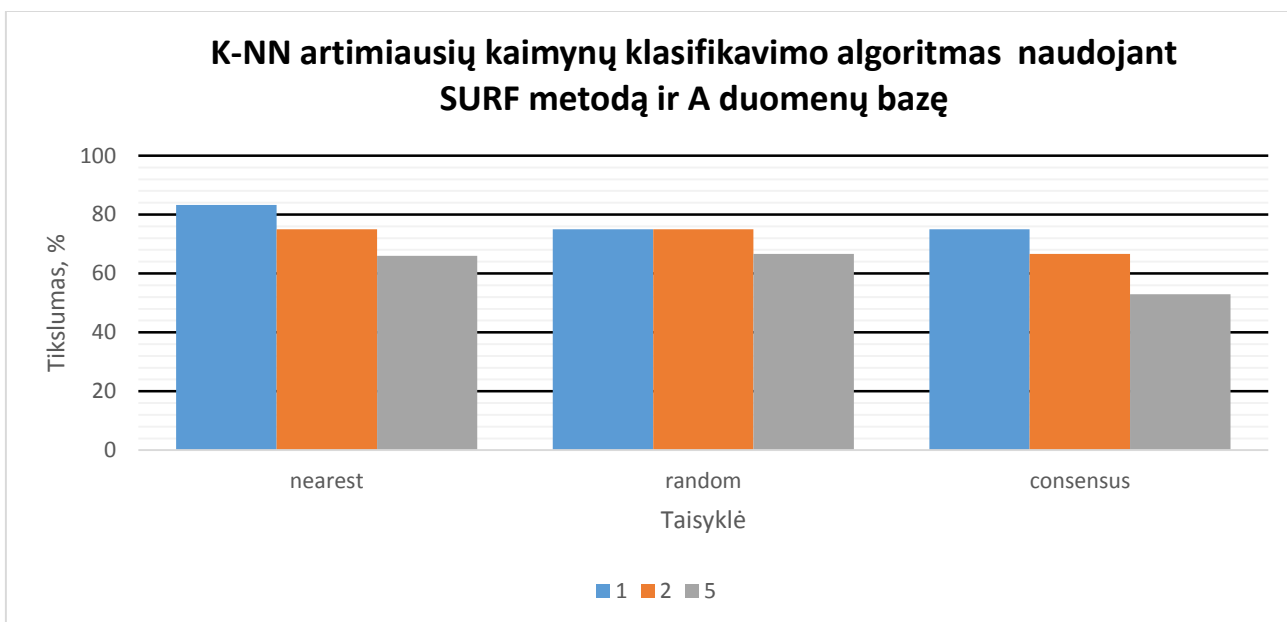
Šiame skyriuje yra parodyti tyrimo rezultatai, kai artimiausio kaimyno atstumo nustatymui yra naudojamas Euklido atstumas. Tyrimo rezultatai yra parodyti žemiau pateiktuose grafikuose.



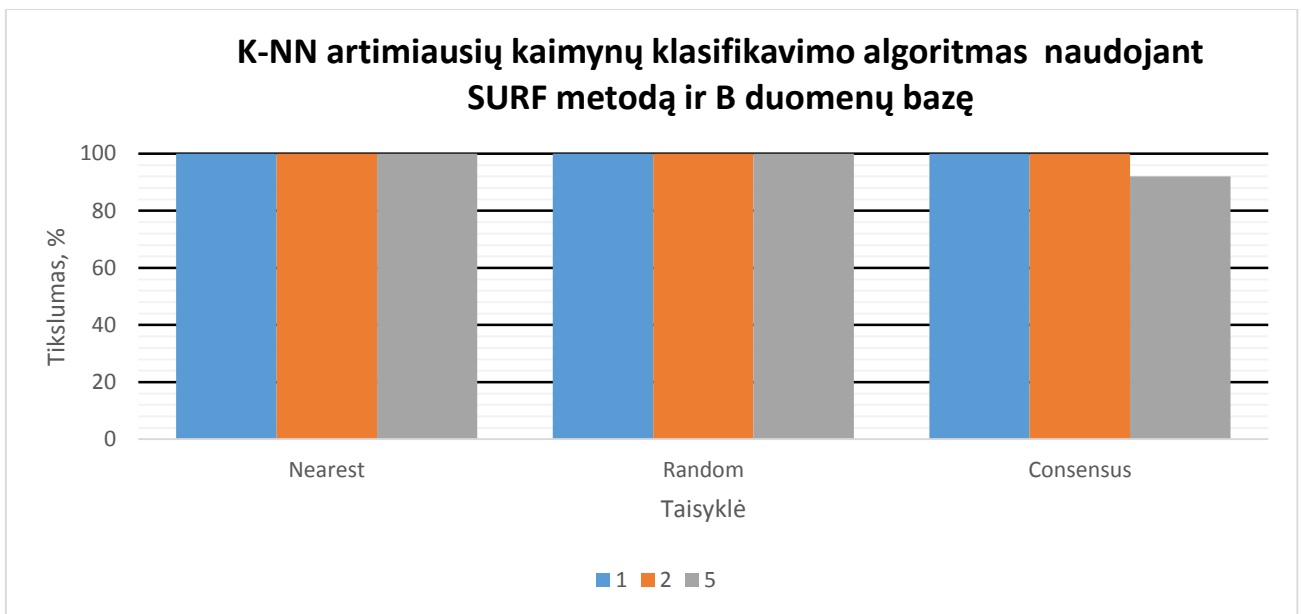
31. Pav. K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant Euklido atstumą, SIFT metodą ir A duomenų bazę



32. Pav. K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant Euklido atstumą, SIFT metodą ir B duomenų bazę



33. Pav. K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant Euklido atstumą, SURF metodą ir A duomenų bazę



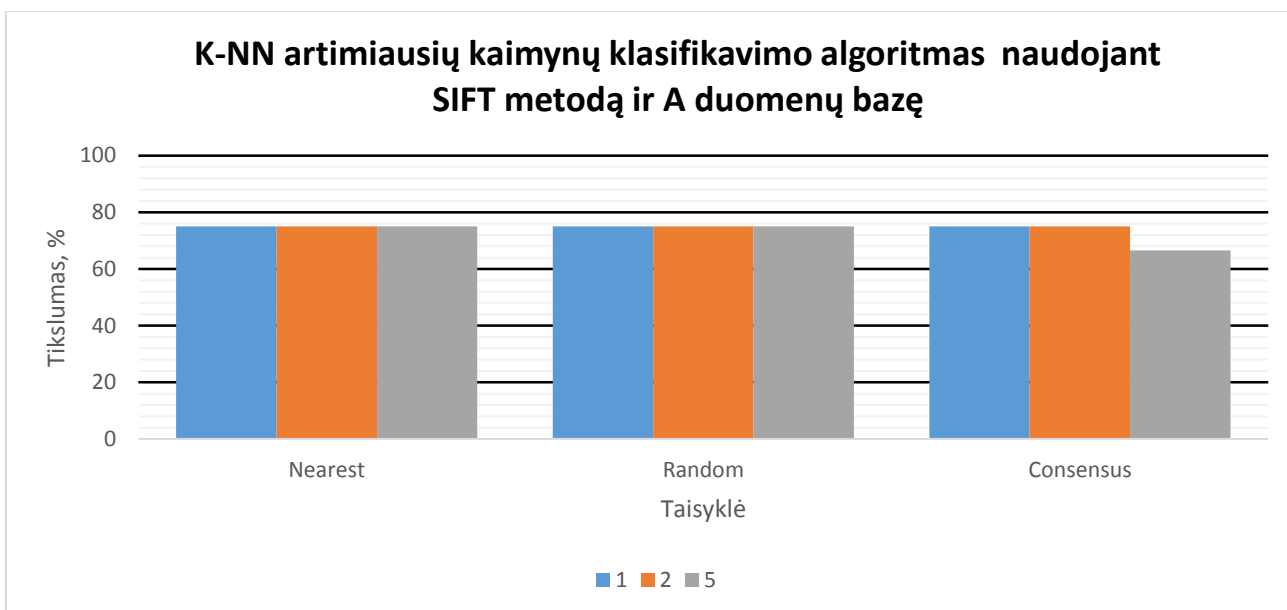
34. Pav. K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant Euklido atstumą, SURF metodą ir B duomenų bazę

Kaip matyti iš aukščiau pateiktų grafikų, nepriklausomai nuo ypatingųjų taškų deskriptorių gavimo būdų, A duomenų bazė pasirodė prasčiau, nei B duomenų bazė. Su A duomenų baze buvo pasiektas maksimalus 83.3% tikslumas, o su B duomenų baze pasiektas 100% atpažinimo tikslumas.

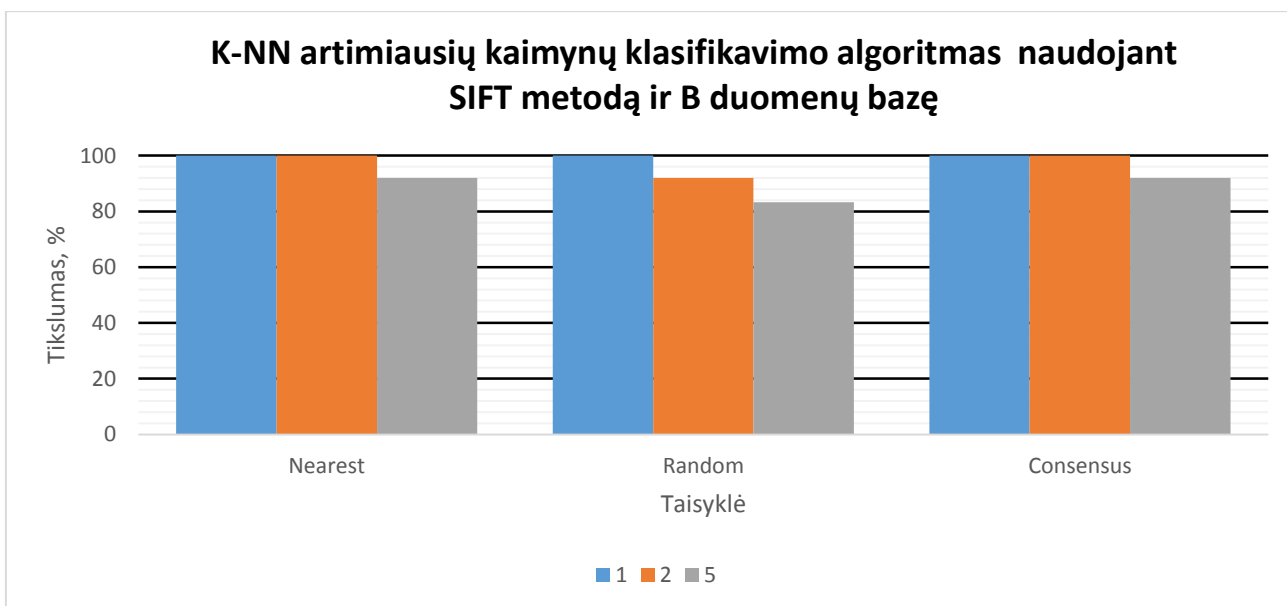
Kaip matyti iš grafikų, algoritme didinant artimiausių kaimynų skaičių k , beveik visur pastebėtas atpažinimo kokybės kritimas, t. y. kuo didesnė k reikšmė, tuo testuojamų nuotraukų atpažinimo kokybė prastesnė.

3.2.2 „Cityblock“ atstumo taisyklė

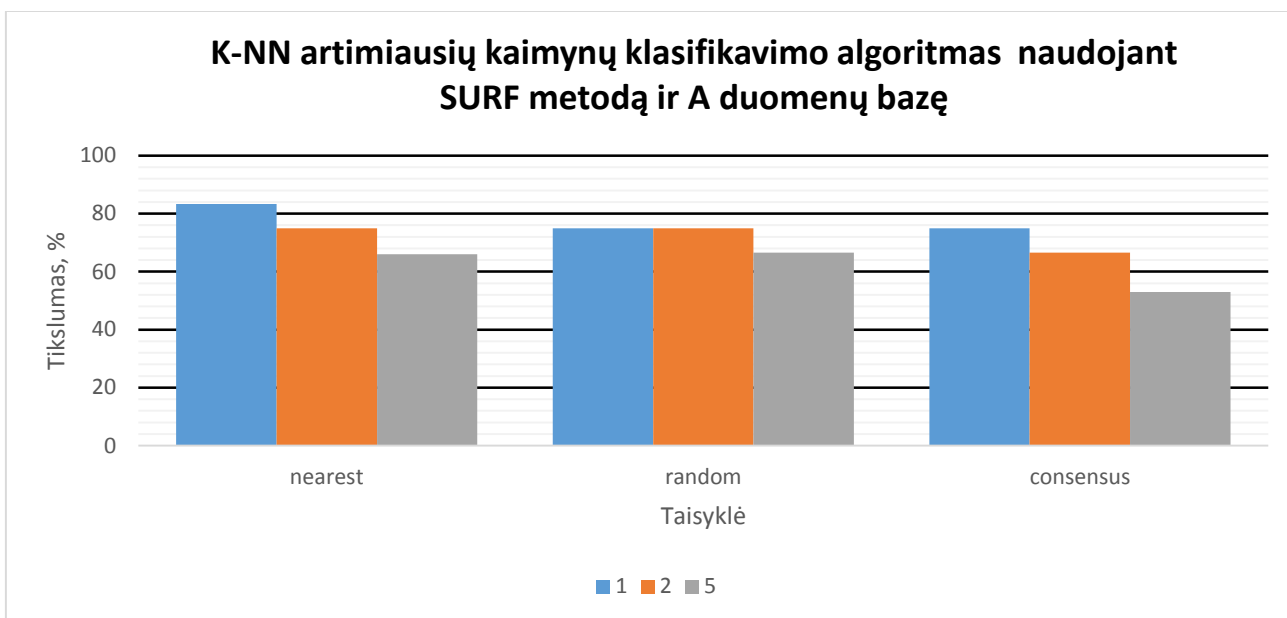
Šiame skyriuje yra parodyti tyrimo rezultatai, kai artimiausio kaimyno atstumo nustatymui yra naudojamas „Cityblock“ atstumo taisyklė. Tyrimo rezultatai yra parodyti žemiau pateiktuose grafikuose.



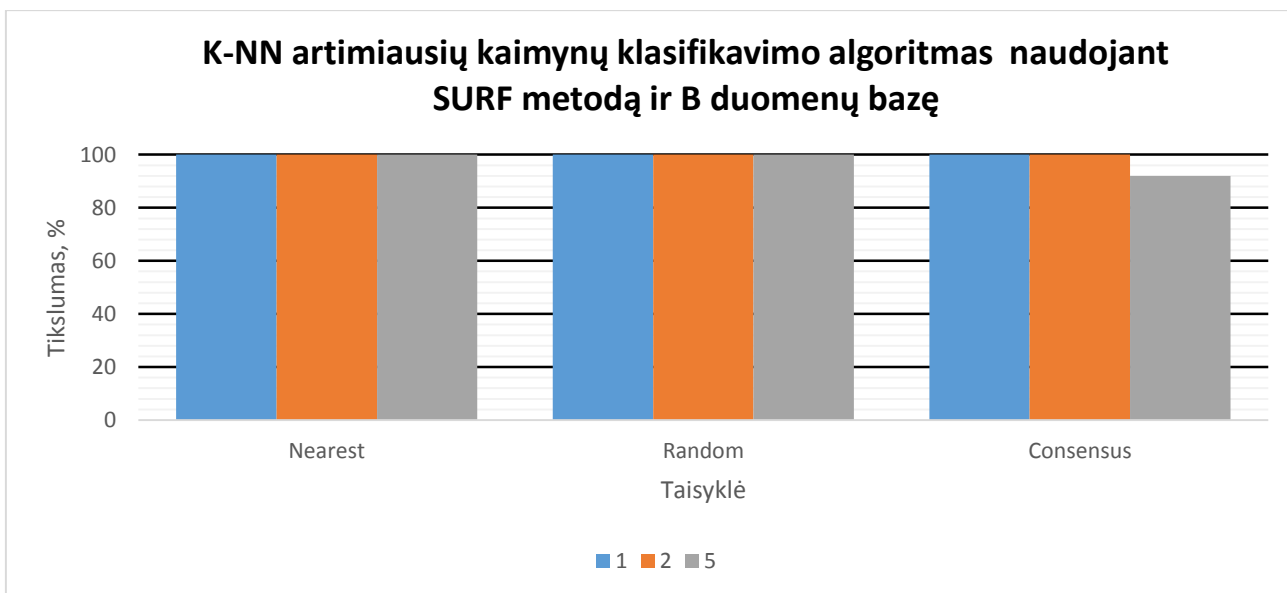
35. Pav. K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant „Cityblock“ atstumą, SIFT metodą ir A duomenų bazę



36. Pav. K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant „Cityblock“ atstumą, SIFT metodą ir B duomenų bazę



37. Pav. K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant „Cityblock“ atstumą, SURF metodą ir A duomenų bazę

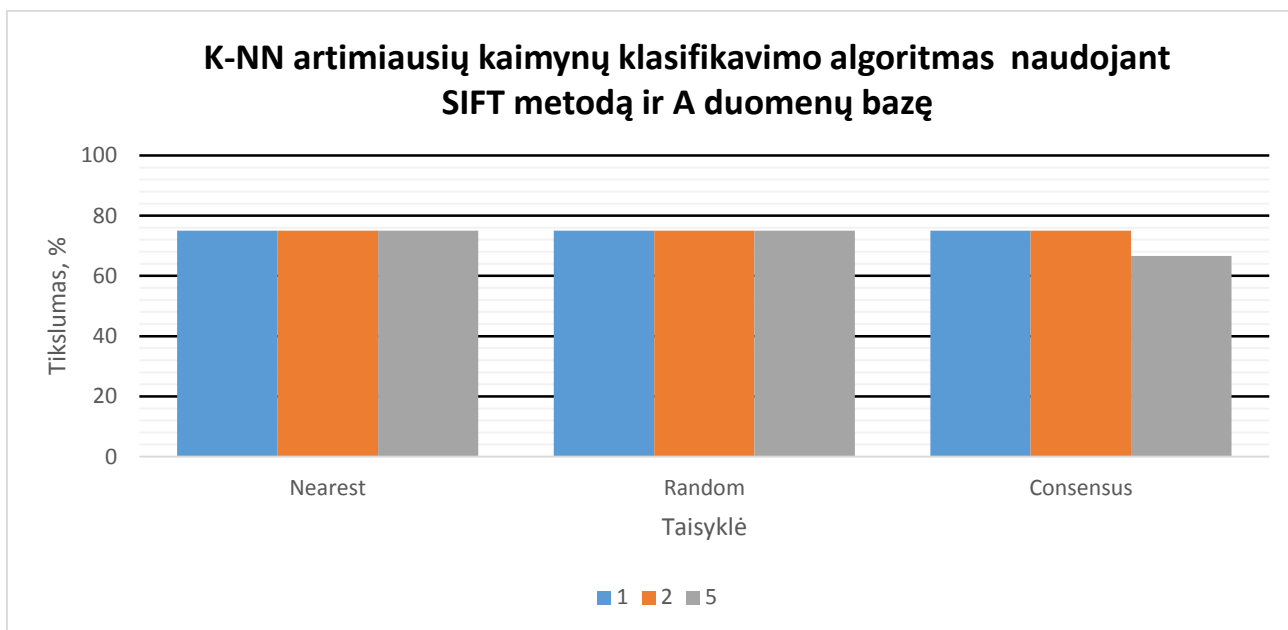


38. Pav. K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant „Cityblock“ atstumą, SURF metodą ir B duomenų bazę

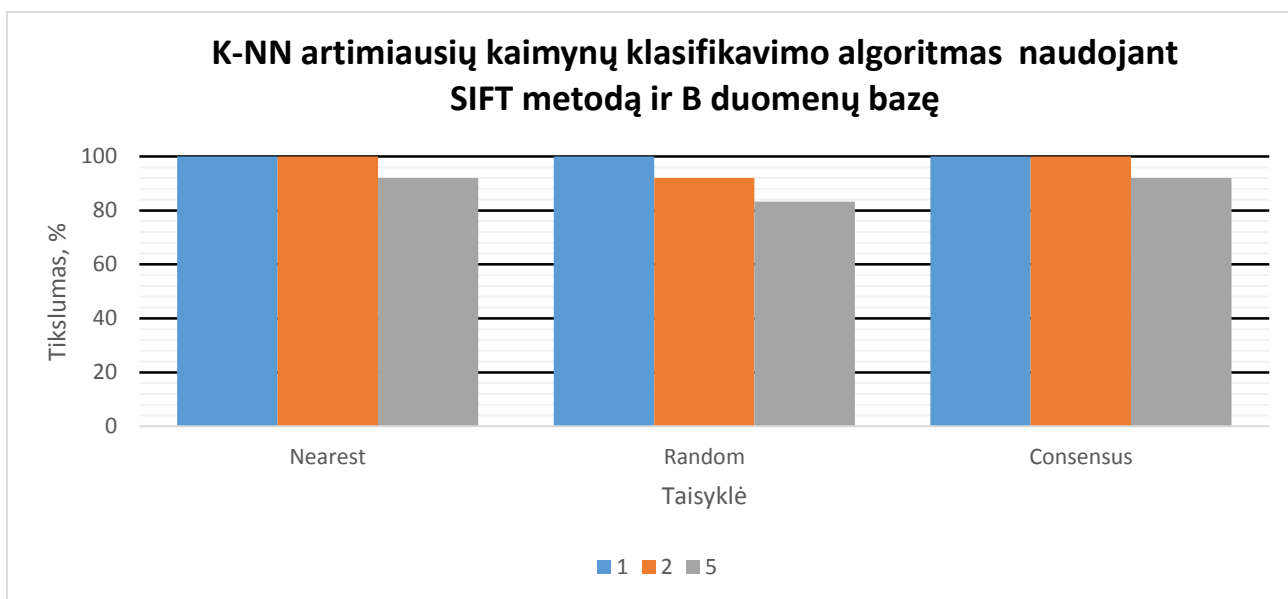
Tyrimo rezultatai parodė, kad naudojant „Cityblock“ atstumą, kaip ir Euklido atstumą, B duomenų bazė pasirodė kur kas geriau nei A duomenų bazė, kur atitinkamai geriausi rezultatai siekė 100% ir 75%.

3.2.3 „Cosine“ atstumo taisyklė.

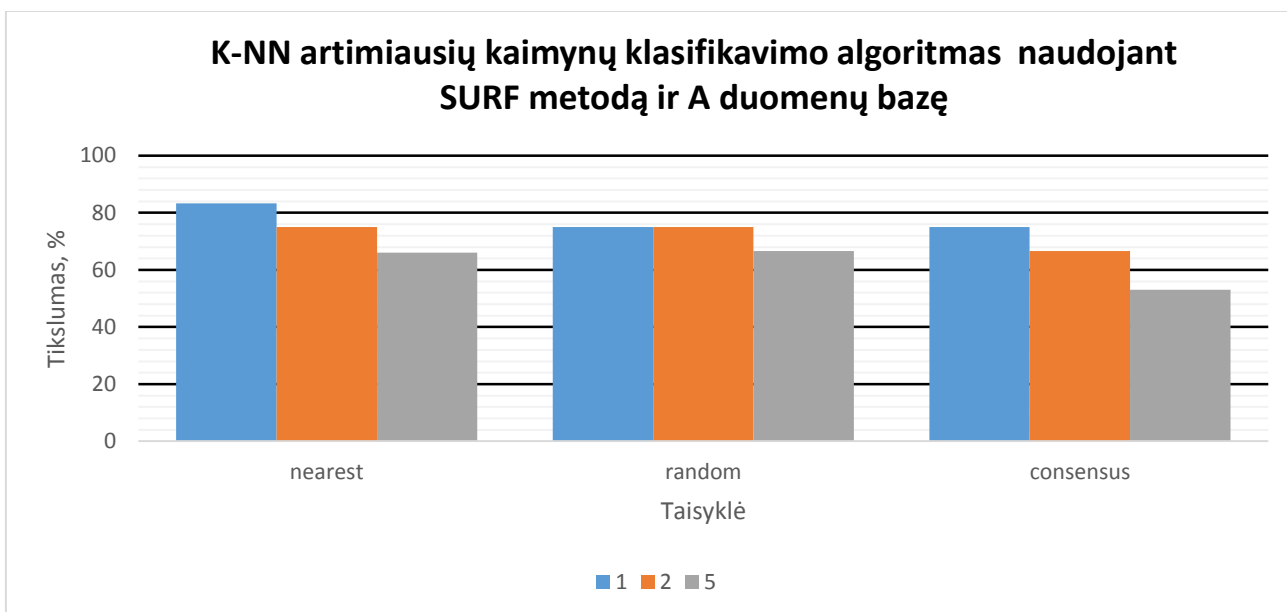
Šiame skyriuje yra apžvelgiami tyrimo rezultatai, kur atstumui taro tiriamo deskriptoriaus ir duomenų bazės deskriptorių nustatymui yra naudojamas „cosine“ atstumas. Žemiau pateiktuose grafikuose yra parodyti tyrimo rezultatai.



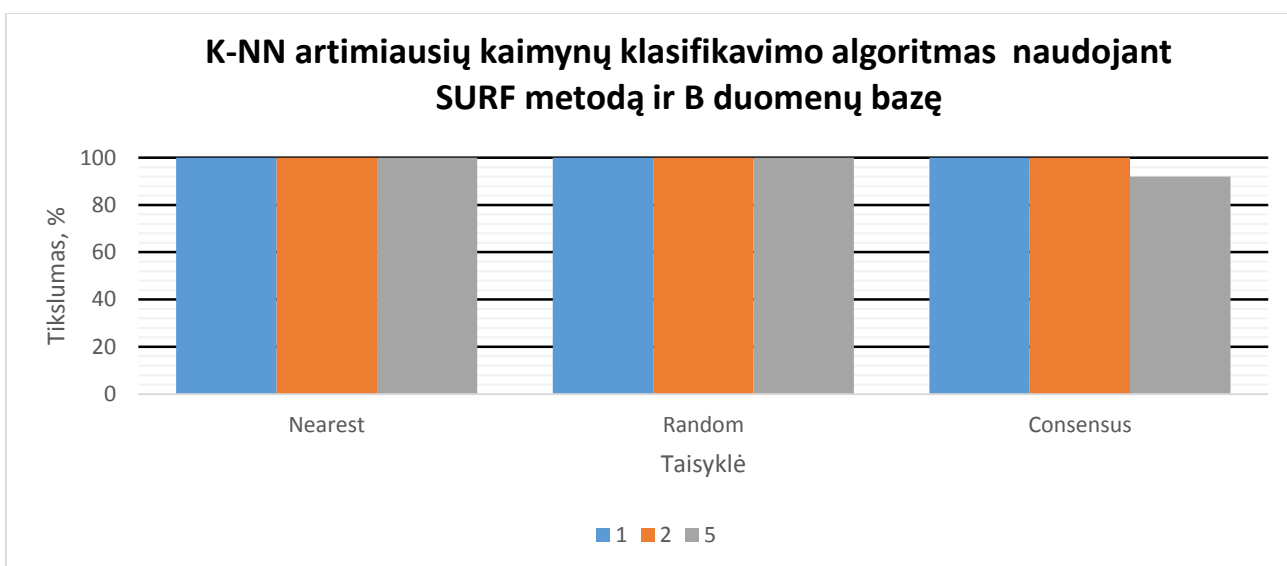
39. Pav. K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant „Cosine“ atstumą, SIFT metodą ir A duomenų bazę



40. Pav. K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant „Cosine“ atstumą, SIFT metodą ir B duomenų bazę



41. Pav. K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant „Cosine“ atstumą, SURF metodą ir A duomenų bazę

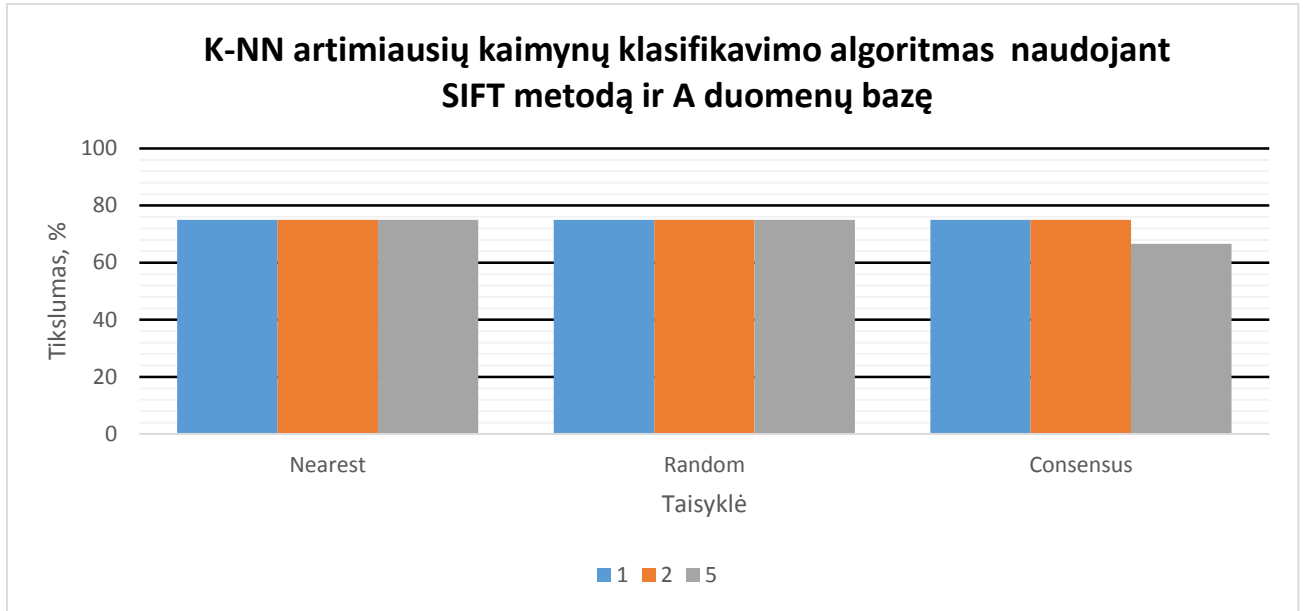


42. Pav. K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant „Cosine“ atstumą, SURF metodą ir B duomenų bazę

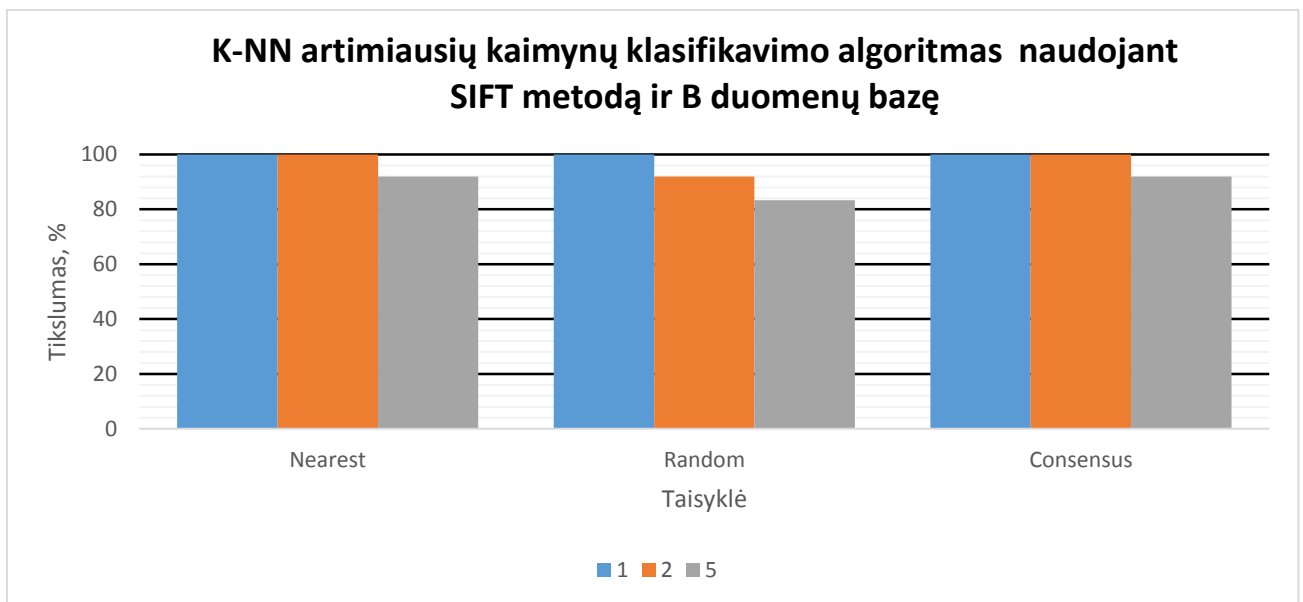
Kaip matyti iš grafikų, naudojant šį metodą, tyrimų rezultatai taip pat parodė gerus rezultatus, tačiau esant skirtingoms algoritmų modifikacijoms, sistema tampa kiek mažiau tiksli ir atspėjimo procentas, krenta. Didinant k reikšmę, atspėjimo kokybė krenta.

3.2.4 Koreliacijos atstumo taisyklė.

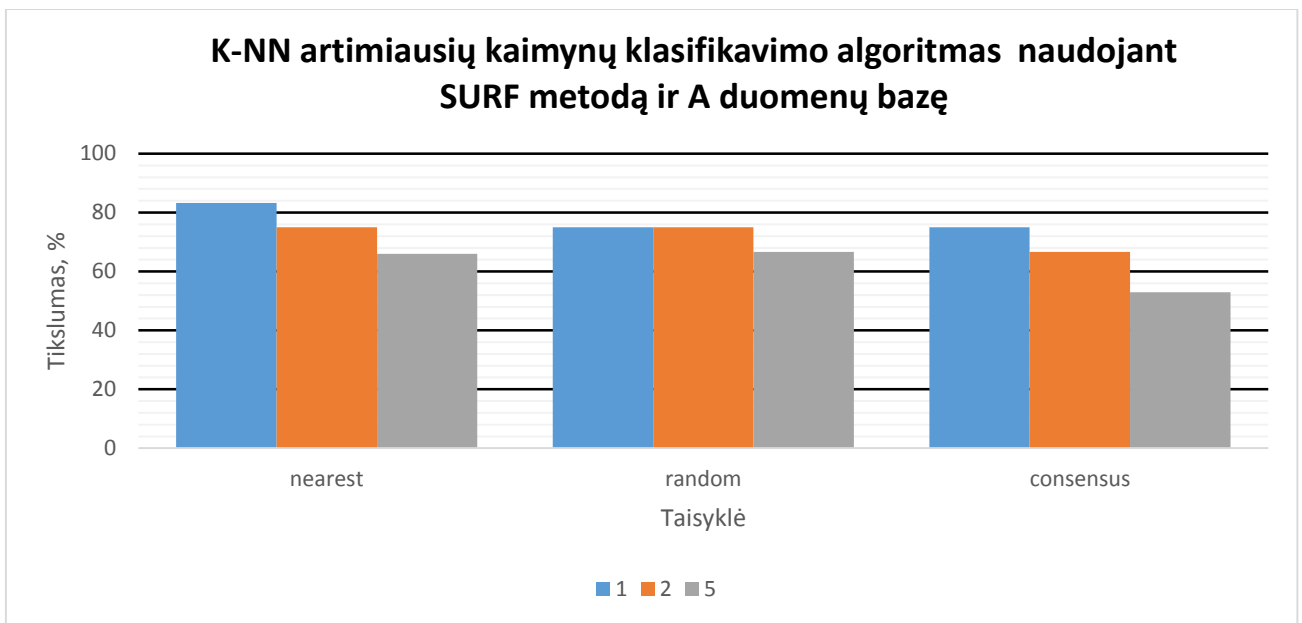
Šiame skyriuje yra parodyti tyrimo rezultatai, kur atstumas tarp tiriamo deskriptoriaus, ir duomenų bazės deskriptorių yra išmatuojamas koreliacijos taisykle. Žemiau pateiktuose grafikuose yra parodyti tyrimo rezultatai.



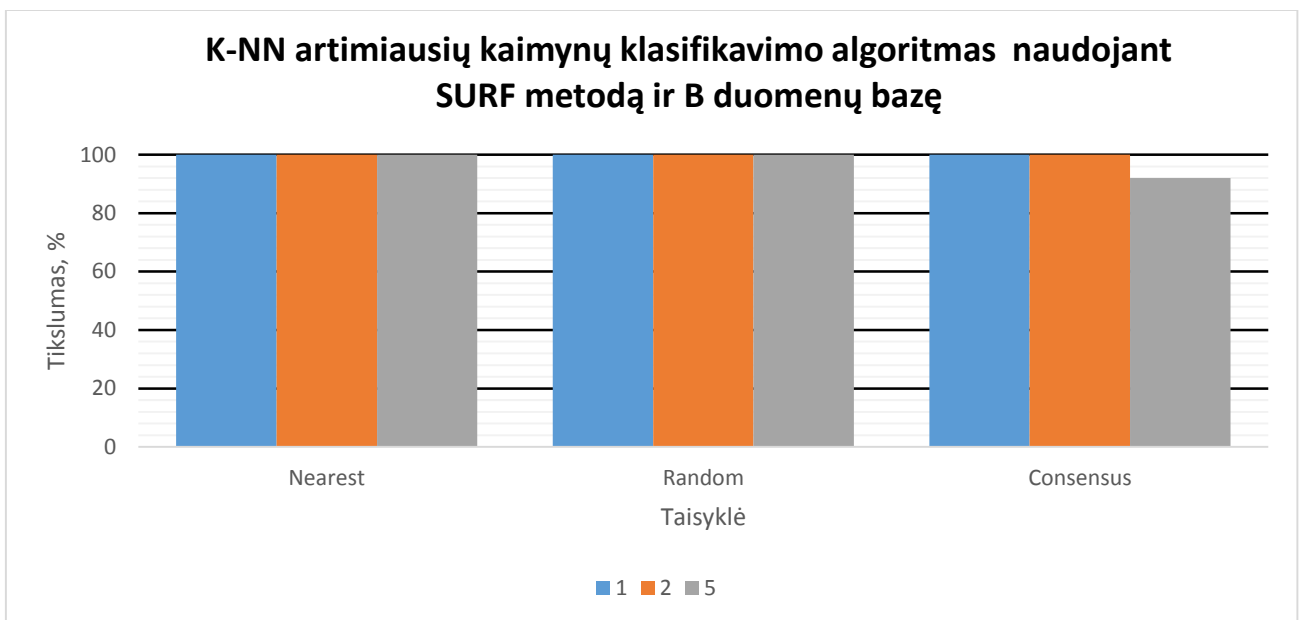
43. Pav. K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant Koreliacijos atstumą, SIFT metodą ir A duomenų bazę



44. Pav. K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant Koreliacijos atstumą, SIFT metodą ir B duomenų bazę



45. Pav. K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant Koreliacijos atstumą, SURF metodą ir A duomenų bazę



46. Pav. K-NN artimiausių kaimynų klasifikavimo algoritmas naudojant Koreliacijos atstumą, SURF metodą ir B duomenų bazę

Šis metodas, lyginant su aukščiau pateiktais tyrimais parodė labai panašius rezultatus. Geriausi rezultatai pasiekti naudojant B duomenų bazę, nepriklausomai nuo metodo naudoti gauti ypatinguosius taškus, ir siekia 100%, o su A duomenų baze geriausias rezultatas pasiektas su SURF metodu, kai artimiausių kaimynų skaičius $k = 1$, naudojant taisyklę „*Nearest*“ ir siekė 83%.

3.2.5 K-NN artimiausių kaimynų metodu pagrįstas klasifikavimo algoritmo skyriaus apibendrinimas

Kaip matyti iš rezultatų, geriausi rezultatai buvo pasiekti su B duomenų baze, kur duomenų bazė yra sudaryta fotografuojant kvėpavimo kaukes nukreiptas į fotoaparato objektyvą 90° kampu, nepriklausomai nuo ypatingųjų taškų gavimo metodo. Atlikus tyrimą pastebėta, kad didinant k skaičių, kvėpavimo kaukės priskyrimo procentas mažėja. Keičiant atstumo nustatymo būdą nuo tiriamo deskriptoriaus su duomenų baze, didelio skirtumo nebuvo pastebėta, ir rezultatas buvo panašus visais tiriamais atvejais. Tačiau tyrimo rezultatai rodo, kad deskriptoriaus priskyrimo kuriai nors klasei, priklausomai nuo taisyklės turi kvėpavimo kaukės priskyrimo tikslumui įtakos. Stabiliausius rezultatus rodė taisyklė „nearest“ (artimiausias tiriamajam deskriptoriui).

Atliekant bandymus taip pat buvo iširta metodų greitaveika. Šie rezultatai yra pateikti 3 ir 4 lentelėje.

3. lentelė. Algoritmo greitaveika, naudojant SIFT metodą ypatingųjų taškų radimui.

Ypatingųjų taškų gavimo metodas	SIFT							
	A				B			
Duomenų bazė								
Naudotas atstumo nustatymo metodas	Euklido	City-block	Cosie	Koreliacijos	Euklido	City-block	Cosie	Koreliacijos
Vidutinis vienos nuotraukos patikrinimo laikas, s	6	6.1	5.5	5.7	6.3	7.1	7.1	6.6

4. lentelė. Algoritmo greitimeika, naudojant SURF metodą ypatingųjų taškų radimui.

Ypatingųjų taškų gavimo metodas	SURF							
Duomenų bazė	A				B			
Naudotas atstumo nustatymo metodas	Euklido	City-block	Cosie	Koreliacijos	Euklido	City-block	Cosie	Koreliacijos
Vidutinis vienos nuotraukos patikrinimo laikas, s	0.9	0.8	0.9	0.9	0.8	0.8	0.9	0.9

Kaip matyti iš 3 lentelės vidutinis vienos nuotraukos patikrinimo greitis naudojant artimiausių kaimynų algoritmą, pagrįstą SIFT ypatingųjų taškų radimo metodu yra ~6s, o iš 4 lentelės matyti kad vienos nuotraukos, naudojant SURF metodą rasti ypatingiesiems taškams, vidutinis patikrinimo laikas yra ~0.8s, nepriklausomai nuo naudojamos duomenų bazės.

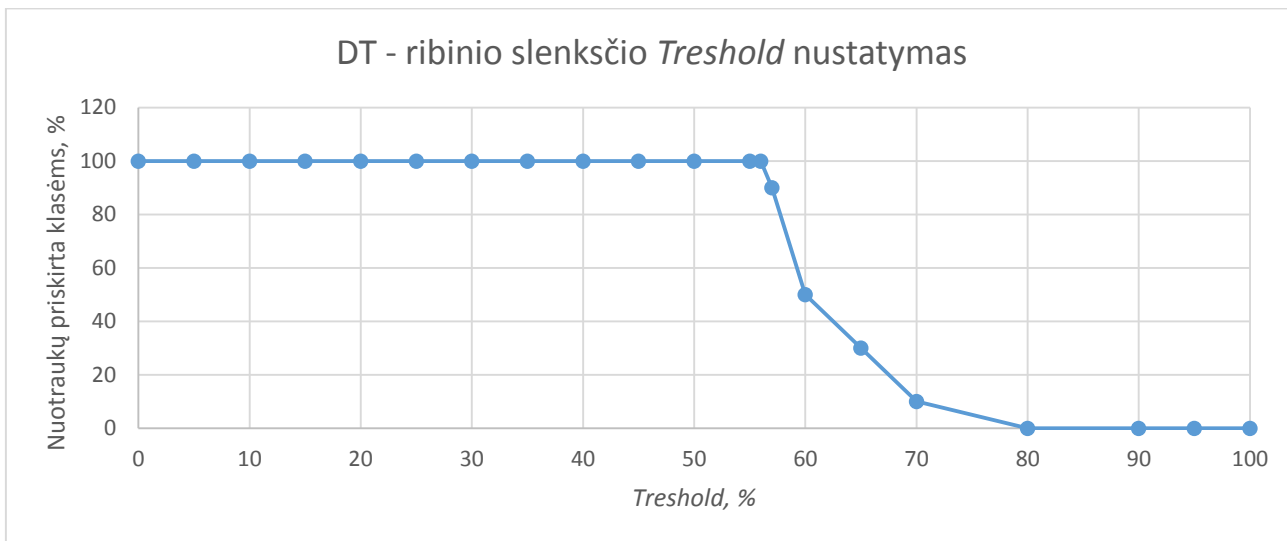
3.3 DT sprendimo medžiu pagrįsto klasifikavimo algoritmo tyrimai

Šiame skyriuje yra parodyti kvėpavimo kaukių klasifikavimo algoritmo tikslumo tyrimai, kai klasifikatorius yra DT – sprendimų medis. Tyrimai bus atliekami atsižvelgiant į du pagrindinius algoritmo parametrus – lapų skaičių (angl. *Minleaf*) ir mazgų (šakų angl. *MinParent*). Atsižvelgus į duomenų bazės dydį, lapų ir mazgų skaičius naudotas tyrime yra toks: 1, 50, 100, 150, 200, 250, 300, 350, 400, 450, 500.

Kaip ir naudojant K-NN artimiausių kaimynų metodą, šis klasifikatorius taip pat kiekvieną tikrinamos nuotraukos deskriptorių priskiria atitinkamai klasei. Todėl norint išvengti neteisingų klasės priskyrimų klasei, pradžia yra nustatoma ribinio slenksčio riba (angl. *Threshold*). Ši dalis yra nustatoma kaip procentas, vienai klasei priskirtų deskriptorių skaičiaus, nuo visų tikrinamos nuotraukos deskriptorių, nes deskriptorių skaičius kiekvienai tiriamai nuotraukai skiriasi.

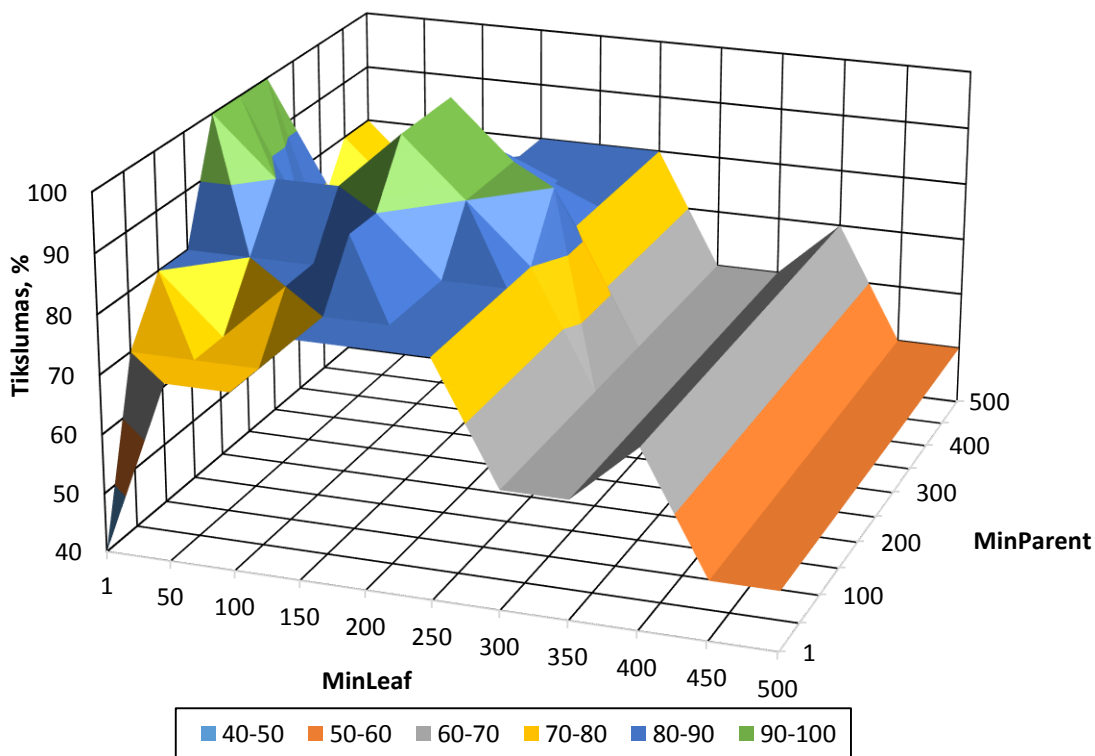
Kaip matyti iš 47 paveikslo, esant ribiniam slenksčiui 56% dar visom tikrinamoms nuotraukoms yra priskiriamos atitinkančiai klasei, o esant šiai ribai 57% tik 90% nuotraukų yra priskiriama atitinkamai klasei. Atsižvelgiant į tyrimo rezultatus, ribinis slenkstis yra parenkama 56%. T. y. jei tikrinamos nuotraukos 56% deskriptorių ir daugiau bus priskirta kuriai nors vienai klasei, tai tada

tikrinamai nuotraukai bus priskirta atitinkama klasė, o priešingu atveju nebus priskiriama nei viena klasei.



47. Pav. Ribinio slenksčio *Threshold* nustatymas DT – sprendimo medžio klasifikatoriui.

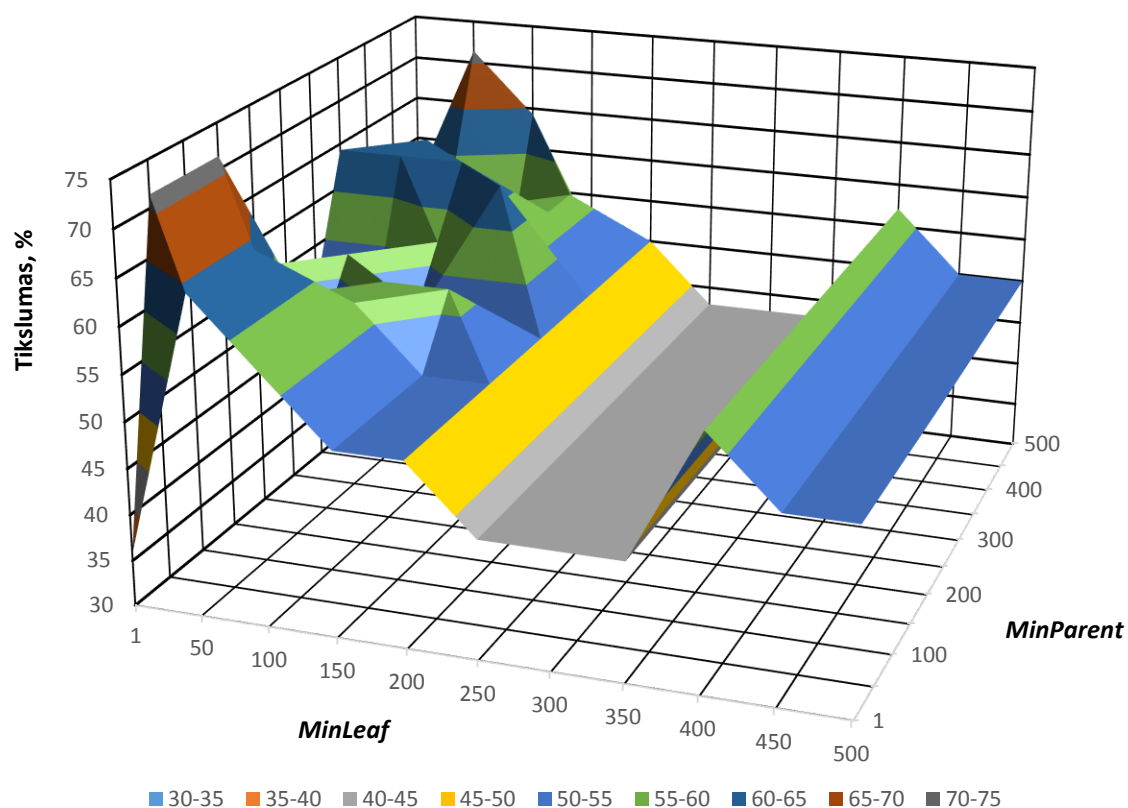
DT - sprendimo medžio klasifikavimo algoritmas su SIFT metodu ir A duomenų baze



48. Pav. DT – sprendimų medžio klasifikatoriaus tikslumo tyrimas, naudojant SIFT metodą ir A duomenų bazę

Aukščiau parodytame tikslumo paviršiuje (žr. 48 paveikslą) yra parodyti tyrimo rezultatai, gauti tyrimą atlikus DT - sprendimo medžio klasifikavimo algoritmu, kur ypatingieji taškai yra gauti naudojant SIFT metodą iš A duomenų bazės. Kaip matyti iš tikslumo paviršiaus, naudojant šį metodą, buvo pasiekti geri rezultatai, kurie siekė 100%. Kaip matyti paviršiuje, geriausi rezultatai buvo užfiksuoti dviejose vietose, kai mažiausias lapų skaičius buvo 1 ir 150, o abiem atvejais mažiausias šakų skaičius buvo 200-300. Tai parodo, kad abiem atvejais buvo sukurta 200-300 taisyklių, kurios padeda nuspręsti, kuriai klasei tikrinamą nuotrauką priskirti.

DT - sprendimo medžio klasifikavimo algoritmas su SIFT metodu ir B duomenų baze

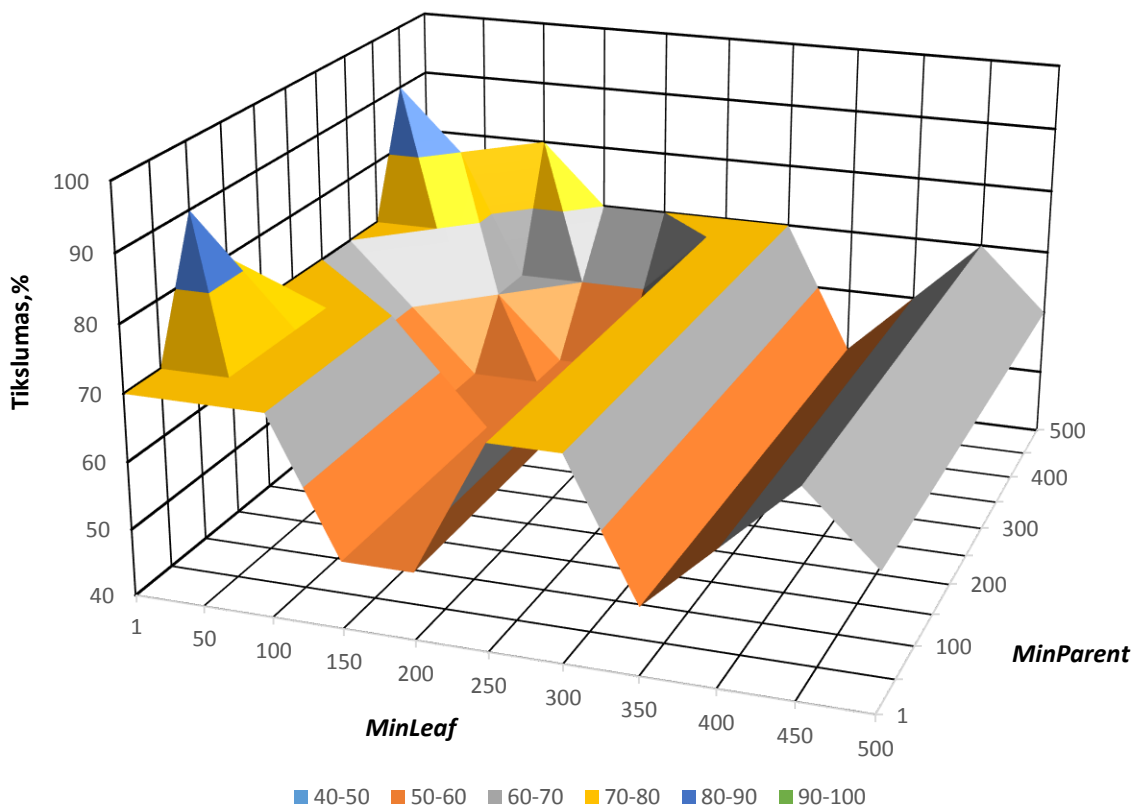


49. Pav. DT – sprendimų medžio klasifikatoriaus tikslumo tyrimas, naudojant SIFT metodą ir B duomenų bazę

49 paveiksle yra parodytas paviršiaus tikslumo grafikas, kur ypatingieji taškai yra gauti SIFT metodu, kaip ir prieš tai atliktame tyrime, tik jie gauti iš B duomenų bazės. Kaip matyti iš paviršiaus, naudojant B duomenų bazę su SIFT metodu gautais ypatingaisiais taškais, bandymų rezultatai parodė prastesnius rezultatus, ir kvėpavimo kaukės klasifikavimo algoritmas parodė tik 71.4% tikslumą, o prasčiausias rezultatas yra tik 35,7%, kai minimalus lapų ir mazgų skaičius yra 1. Pastebėta, kad kai

didėja minimalus lapų skaičius, tiek su A duomenų baze ir tiek su B duomenų baze, rezultatai prastėja, ir neduoda gerų rezultatų. O minimaliam lapų skaičiui esant mažam, tyrimo rezultatai rodo didesnę tikslumą.

DT - sprendimo medžio klasifikavimo algoritmas su SURF metodu ir A duomenų baze

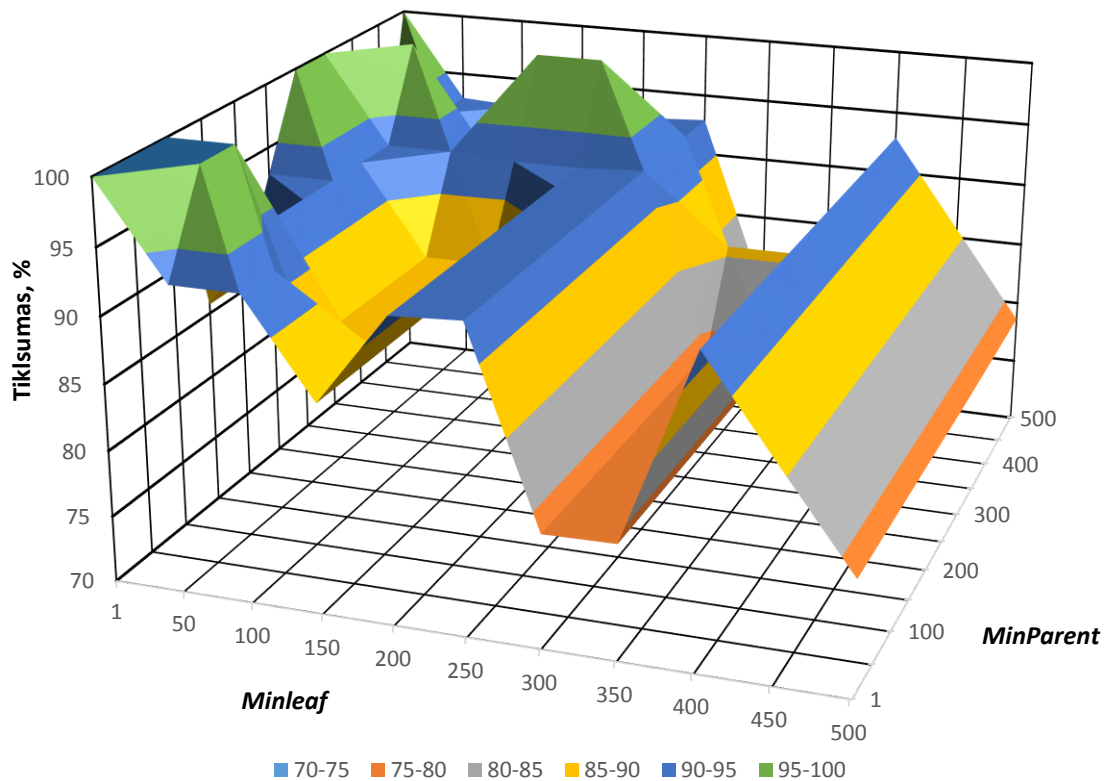


50. Pav. DT – sprendimų medžio klasifikatoriaus tikslumo tyrimas, naudojant SURF metodą ir A duomenų bazę

Aukščiau parodytame paviršiuje parodyti sprendimo medžio algoritmo tyrimo rezultatai, kai ypatingieji taškai iš A duomenų bazės yra gauti naudojant SURF metodą. Lyginant SIFT metodą su A duomenų baze, rezultatai yra prastesni, ir atpažinimo tikslumas tesiekia tik 90%, kai minimalus lapų skaičius algoritme yra 1, o šakų skaičius 50 ir 450. Prasčiausi rezultatai yra kai minimalus lapų skaičius yra 100-150 ir 350.

Tiriant algoritmą su SURF gautais ypatingaisiais taškais iš B duomenų bazės (žr. 51 paveikslą) buvo gauti geriausi rezultatai ir atpažinimo tikslumas siekia 100%. Šie rezultatai yra pasiekiami su keliomis algoritmo parametrų reikšmėmis.

DT - sprendimo medžio klasifikavimo algoritmas su SURF metodu ir B duomenų baze



51. Pav. DT – sprendimų medžio klasifikatoriaus tikslumo tyrimas, naudojant SURF metodą ir B duomenų bazę

Lyginant sprendimų medį su artimiausių kaimynų algoritmu, artimiausių kaimynų algoritmas rodė geresnius rezultatus nei sprendimų medis, bet jo greیتaveika yra kur kas mažesnė. Sprendimo medžio greیتaveikos rezultatai yra parodyti 5 lentelėje.

5. lentelė. DT – sprendimo medžio greیتaveika.

Ypatingųjų taškų gavimo metodas	SIFT		SURF	
	A	B	A	B
Duomenų bazė				
Vidutinis vienos nuotraukos patikrinimo laikas, s	0.12	0.16	0.11	0.1

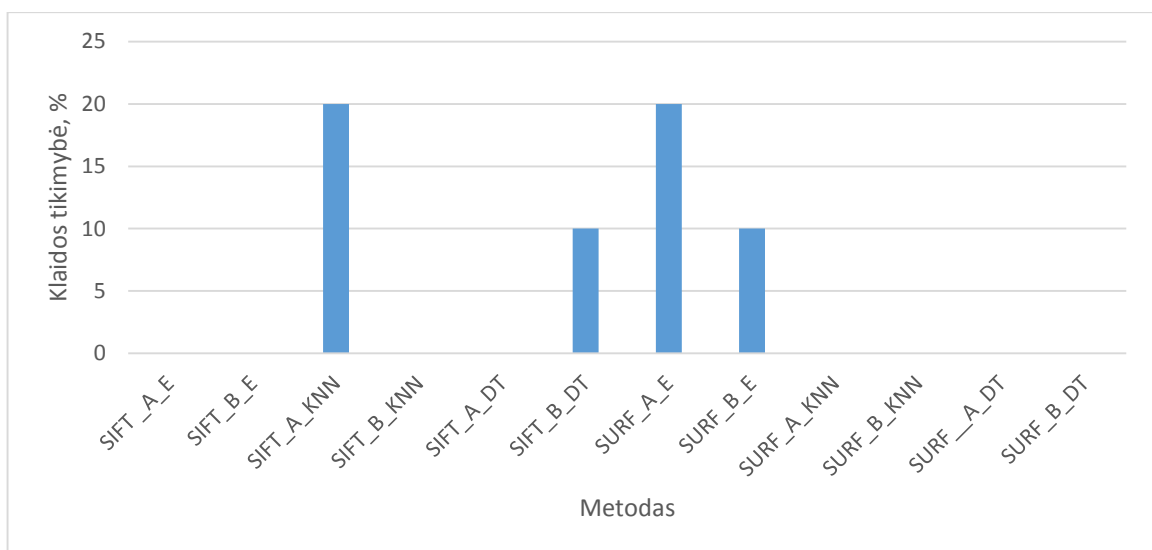
Kaip matyti iš lentelės vienos nuotraukos patikrinimo laikas yra ~0.12s ir nepriklauso nuo metodo naudoto gauti ypatingiesiems taškams, o tai yra pasiekama dėl to, kad pradžioje yra

sukuriamas sprendimo medžio modelis, o vėliau yra tikrinamos reikšmės, priešingai nei artimiausių kaimynų algoritme, ko pasekoje ir pasiekama tokia greitaveika.

3.4 Klaidingai teigiamas, klaidingai neigiamas testas

Kadangi šių algoritmų tikslas yra klasifikuoti tik kvėpavimo kaukes, bei siekiant išsiaiškinti ar algoritmas veikia tinkamai ir klasifikuoja tik kvėpavimo kaukes yra atliekamas „Klaidingai teigiamas, Klaidingai neigiamas (angl. *False positive, False negative*) testas, kurio metu siekiama išsiaiškinti, kaip algoritmas klasifikuoja visiškai su šiuo magistriniu darbu nesusijusių nuotraukų klasifikavimu. Tam buvo sudaryta tikrinamų nuotraukų imtis, kurioje yra nuotraukos su skirtingais fonais, ir vaizdais jose, pavyzdžiui akmuo, ar žmogaus ranka. Šis testas laikomas pasisekusių, jei tikrinamos nuotraukos nėra priskiriamos jokiai klasei ir yra atmetamos.

Šis testas yra atliekamas su visais algoritmais, kur jų parametrai yra paimami iš gautų rezultatų, kur klasifikavimo rezultatai buvo geriausi. Tyrimo rezultatai yra parodyti 52 paveiksle.



52. Pav. Klaidingai teigiamas, Klaidingai neigiamas testo rezultatai

Grafike yra pavaizduota Klaidingai teigiamo, Klaidingai neigiamo testo rezultatai, kurie parodo, kokia yra klaidos tikimybė. Kaip matyti iš grafiko, klaidos tikimybė yra tik 4 metodais. Klaidingai priskirtos kaukės buvo su artimiausiu kaimyno klasifikatoriumi naudojant SIFT metodą gauti ypatinguosius taškus su A duomenų baze 20%, su sprendimo medžio klasifikatoriumi ir SIFT metodu gautais ypatingaisiais taškais gautais iš B duomenų bazės 10%, su Euklido klasifikatoriumi naudojant SURF metodą gauti iš ypatingųjų taškų su A duomenų baze 20 ir su B% duomenų baze 10%.

4 IŠVADOS IR REZULTATAI

1. Atlikus literatūros analizę, kvėpavimo kaukės klasifikavimui buvo pasirinkti ypatingųjų taškų gavimui iš testuojamos nuotraukos SIFT ir SURF metodai, tam, kad pilnavertiškai ištirti algoritmą. Kvėpavimo kaukės klasifikavimui buvo pasirinkti trys skirtingi metodai: Euklido atstumas, K-NN artimiausių kaimynų klasifikatorius ir DT – sprendimo medžio klasifikatorius.
2. Atlikus ypatingųjų taškų skaičiaus priklausomybę nuo nuotraukos dydžio buvo pasirinktas nuotraukos dydis 800x600, nes su šiuo nuotraukos dydžiu yra gaunama pakankamai, bet ne per daug, ypatingųjų taškų, tam kad kokybiškai klasifikuoti kvėpavimo kaukes. Su SIFT metodu jų vidutinis skaičius iš vienos nuotraukos yra apie ~400, o su SURF metodu jų yra apie ~115.
3. Atlikus eksperimentus buvo nustatyti geriausi parametrai kiekvienai algoritmo modifikacijai. Jie parodyti 6 lentelėje.
4. Buvo atliktas Klaidingai teigiamas, Klaidingai neigiamas testas, tam kad nustatyti kiekvieno algoritmo, su geriausias tyrimuose pasirodžiusiais rezultatais, klaidos tikimybę. Tyrimo rezultatai yra parodyta 6 lentelėje skiltyje: *Klaidos tikimybė*.
5. Išanalizavus literatūrą, buvo nustatyta, kad SIFT metodu gautų deskriptorių ilgis yra 128 simboliai, o SURF metodu gautų deskriptorių ilgis yra 64 simboliai. Atlikus eksperimentinius bandymus, buvo nustatyta, kad naudojant SURF metodu gautus ypatinguosius taškus kvėpavimo kaukės klasifikavimui, pasiekama didesnė greitaveika, nepriklausomai nuo klasifikavimo algoritmo.
6. Visi eksperimentiniai tyrimai buvo atliekami su dviem duomenų bazėmis, A – nuotraukos gautos 45° laipsnių kampu į fotoaparato objektyvą ir B – nuotraukos gautos 90° laipsnių kampu į fotoaparato objektyvą, tam, kad nustatyti duomenų bazės įtaką klasifikavimo algoritmo kokybei. Atlikus visus eksperimentinius tyrimus buvo nustatyta, kad naudojant A duomenų bazę atpažinimo vidurkis yra 91.3%, o naudojant B duomenų bazę atpažinimo vidurkis yra 95.2%. Todėl yra daroma išvada, kad kvėpavimo kaukės klasifikavimui labiau tinkama yra B duomenų bazė.
7. Iš gautų eksperimentinių tyrimų rezultatų, geriausias algoritmo derinys klasifikuoti kvėpavimo kaukes yra: SURF metodu gauti ypatingieji taškai iš B duomenų bazės su sprendimo medžio klasifikatoriumi.

6. lentelė. Tyrimo rezultatų palyginimas.

Ypatingųjų taškų gavimo algoritmas	Klasifikatorius	Duomenų bazė:	Geriausi parametrai:	Tikslumas:	Greitaveika:	Klaidos tikimybė
SIFT	Euklido atstumas	A	<i>distRatio: 0.5 Trashold: 45</i>	100%	60.2s	0%
SIFT	Euklido atstumas	B	<i>distRatio: 0.7 Trashold: 60</i>	100%	50.3s	0%
SIFT	K-NN	A	<i>Trashold: 55% Atstumas: Cosine Taisyklė: Nearest</i>	75%	5.5s	20%
SIFT	K-NN	B	<i>Trashold: 55% Atstumas: Koreliacija Taisyklė: Nearest</i>	100%	6.6s	0%
SIFT	DT	A	<i>Trashold: 56% Minleaf: 1 MinParent: 300</i>	100%	0.12s	0%
SIFT	DT	B	<i>Trashold: 56% Minleaf: 50 MinParent: 1</i>	71.4%	0.16s	10%
SURF	Euklido atstumas	A	<i>distRatio: 0.4 Trashold: 15</i>	100%	30.3s	20%
SURF	Euklido atstumas	B	<i>distRatio: 0.5 Trashold: 15</i>	100%	28.7s	10%
SURF	K-NN	A	<i>Trashold: 55% Atstumas: Cityblock Taisyklė: Nearest</i>	83.3%	0.8s	0%
SURF	K-NN	B	<i>Trashold: 55% Atstumas: Euklido Taisyklė: Nearest</i>	100%	0.8s	0%
SURF	DT	A	<i>Trashold: 56% Minleaf: 1 MinParent: 100</i>	90%	0.11s	0%
SURF	DT	B	<i>Trashold: 56% Minleaf: 1 MinParent: 1</i>	100%	0.1s	0%

5 LITERATŪROS SĄRAŠAS

1. Mark S. Nixos, Alberto S. Aguado, „Feature Extraction & Image Processing for Computer Vision“ Third edition, 2012 ,Chapter 1, pp 1 – 35 .
2. David G. Lowe, „Distinctive Image Features from Scale-Invariant Keypoints“, International Journal of Computer Vision 60(2), 91-110, 2004
3. H. Bay, A. Ess, T. Tuytelaars, L. V. Gool „Speeded-Up Robust Features (SURF)“ Computer Vision and Image Understanding 110, pp 346-359 2008
4. C. Harris & M. Stephens „A Combined Corner and Edge Detector“ 1988.
5. J. Canny „A Computational Approach to Edge Detection“ pp 679-698, 1986.
6. B.L. Shivakumar, Dr. S. Santhosh Baboo „Detection of Region Duplication Forgery in Digital Images Using SURF“ International Journal of Computer Science Issues, Vol 8 pp 199-205 2014
7. P. Cunningham, S. J. Delany „k-nearest Neighbour Classifiers“ Technical report UCD-CSI-2007-4, 2007.
8. K.Q. Weinberger, J. Blitzer, L. K. Saul „Distance Metric Learning for large Margin Nearest Neighbor Classification“
9. R. M.C.R. de Souza, F. De A.T. de Carvalho „Clustering of interval data based on city-block distances“ Pattern Recognitions Letters 25,pp 353-365 2003
10. L. Wang, Y. Zhang, J. Feng „On the Euclidean Distance of Images“, IEEE transaction on Pattern analysis and machine intelligence, Vol 27, No8, pp 1334-1339, 2005.
11. Srivastava, E. Hong Han, V. Kumar, V. Singh „Parallel Formulations of Decision-Tree Classification Algorithms“ High Performance data Mining, pp237-261, 2002
12. J.R. Quinlan „Introduction of Decision Trees“, Machine Learning pp 81-106, 1986
13. Abdelhalim, I. Traore,, „A new Method for Learning Decision Tree from Rules“. 2009 International Conference on machine Learning and Applications, pp 693-698, 2009