

**Kauno technologijos universitetas**

Informatikos fakultetas

**Trumpalaikio kritulių prognozavimo pagal meteorologinio radaro  
duomenis Lietuvoje tyrimas**

Baigiamasis magistro projektas

---

**Aivaras Čiurlionis**

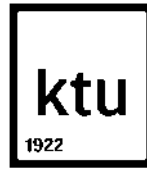
Projekto autorius

**doc. dr. Mantas Lukoševičius**

Projekto vadovas

---

Kaunas, 2020



**Kauno technologijos universitetas**

Informatikos fakultetas

**Trumpalaikio kritulių prognozavimo pagal meteorologinio radaro  
duomenis Lietuvoje tyrimas**

Baigiamasis magistro projektas

Programų sistemų inžinerija (6211BX011)

---

**Aivaras Čiurlionis**

Projekto autorius

**doc. dr. Mantas Lukoševičius**

Projekto vadovas

**doc. dr. Evaldas Vaičiukynas**

Recenzentas

---

Kaunas, 2020

# Trumpalaikio kritulių prognozavimo pagal meteorologinio radaro duomenis Lietuvoje tyrimas

## Akademinio sąžiningumo deklaracija

Patvirtinu, kad mano, Aivaro Čiurlionio, baigiamasis projektas tema „Trumpalaikio kritulių prognozavimo pagal meteorologinio radaro duomenis Lietuvoje 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.

*Vardas, Pavardė:* **Aivaras Čiurlionis**

*Parašas:*

# Turinys

<b>1</b>	<b>Ižanga</b>	<b>10</b>
1.1	Projekto naujumas ir aktualumas . . . . .	10
1.2	Tikslas ir uždaviniai . . . . .	10
1.3	Dokumento struktūra . . . . .	11
<b>2</b>	<b>Analitinė dalis</b>	<b>12</b>
2.1	Analitinės dalies įvadas . . . . .	12
2.2	Kritulių susidarymo ir judėjimo principai . . . . .	13
2.2.1	Kritulių susidarymas . . . . .	13
2.3	Meteorologinis radaras . . . . .	15
2.3.1	Doplerio radaro technologija . . . . .	15
2.3.2	Meteorologinis radaras Lietuvoje . . . . .	16
2.3.3	Radaro ribotumas . . . . .	17
2.4	Trumpalaikės prognozės . . . . .	18
2.5	Mašininis mokymasis ir dirbtiniai neuroniniai tinklai . . . . .	19
2.5.1	Mašininio mokymosi tipai . . . . .	20
2.5.2	Klasifikacija ir regresija . . . . .	20
2.5.3	Dirbtiniai neuroniniai tinklai . . . . .	21
2.6	Rekurentiniai neuroniniai tinklai . . . . .	22
2.7	Mašininio mokymosi taikymas meteorologijoje orų prognozavime . . . . .	23
2.8	Mašininio mokymosi taikymas vaizdo įrašų prognozavime . . . . .	27
<b>3</b>	<b>Projektinė dalis</b>	<b>29</b>
3.1	Sistemos funkcinis aprašymas . . . . .	29
3.1.1	Sistemos funkcijos . . . . .	29
3.1.2	Sistemos kontekstas . . . . .	30
3.2	Sistemos apribojimai . . . . .	31
3.3	Statinis sistemos vaizdas . . . . .	32
3.3.1	Sistemos sudedamosios dalys . . . . .	32
3.3.2	Sistemos duomenų modelis . . . . .	33
3.4	Dinaminis sistemos vaizdas . . . . .	34
3.5	Sistemos diegimas . . . . .	36
3.6	Projektinės dalies rezultatai . . . . .	36
<b>4</b>	<b>Tyrimas</b>	<b>38</b>
4.1	Duomenys . . . . .	38
4.1.1	Duomenų surinkimas . . . . .	38
4.1.2	Duomenų apdorojimas . . . . .	40
4.1.3	Duomenų saugojimas su retosiomis matricomis . . . . .	41
4.2	Duomenų tyrinėjimas . . . . .	42
4.2.1	Kritulių kiekis ir padengimas . . . . .	42
4.2.2	Kritulių judėjimo krypties tendencijos . . . . .	43

4.2.3	Kritulių judėjimo greitis . . . . .	44
4.3	Prognozių tikslumo matavimo metodai . . . . .	45
4.3.1	Šaknis iš vidutinės kvadratinės paklaidos . . . . .	46
4.3.2	<i>HK</i> įvertis . . . . .	46
4.3.3	Dalinis tikslumo įvertis . . . . .	47
4.3.4	Prognozių tikslumo matavimo metodų apibendrinimas . . . . .	49
<b>5</b>	<b>Eksperimentas</b>	<b>50</b>
5.1	Eksperimento sąlygos . . . . .	50
5.1.1	Modelių apmokymo metodas . . . . .	50
5.1.2	Modelio apmokymo parametrai . . . . .	50
5.1.3	Prognozių generavimo ir jų validavimo procesas . . . . .	50
5.2	Eksperimentai su deterministiniais modeliais . . . . .	51
5.3	Prognozės modelių architektūros, besiremiančios neuroniniais tinklais . . . . .	52
5.3.1	3-jų sluoksnių CNN . . . . .	52
5.3.2	4-ių sluoksnių CNN . . . . .	53
5.3.3	Autoenkoderio architektūra . . . . .	53
5.3.4	<i>Conv-LSTM</i> architektūra . . . . .	54
5.4	Papildomo reljefo ir atsitiktinio duomenų kanalų efektyvumo tyrimas 3-jų sluoksnių konvoliuciniame neuroniniame tinkle . . . . .	54
5.4.1	Eksperimento aprašymas . . . . .	54
5.4.2	Eksperimento eiga . . . . .	55
5.4.3	Eksperimento rezultatai . . . . .	56
5.5	Papildomų reljefo ir atsitiktinio duomenų kanalų kombinacijų tyrimas . . . . .	59
5.5.1	Eksperimento aprašymas . . . . .	59
5.5.2	Eksperimento eiga . . . . .	59
5.5.3	Eksperimento rezultatai . . . . .	59
5.6	Išsamus skirtingų architektūrų neuroninių tinklų prognozių generavimo tyrimas . . . . .	60
5.6.1	Eksperimento aprašymas . . . . .	60
5.6.2	Eksperimento eiga . . . . .	61
5.6.3	4-ių sluoksnių CNN architektūros rezultatai . . . . .	61
5.6.4	Autoenkoderio rezultatai . . . . .	62
5.6.5	Konvoliucinio LSTM tinklo rezultatai . . . . .	62
5.6.6	Geriausi modeliai pagal RMSE . . . . .	63
5.6.7	Geriausi modeliai pagal <i>HK</i> įvertį . . . . .	64
5.6.8	Geriausi modeliai pagal FSS įvertį . . . . .	65
5.7	Eksperimentų rezultatų įvertinimas . . . . .	66
5.8	Projekto testinumas . . . . .	67
<b>6</b>	<b>Išvados</b>	<b>68</b>
	<b>Priedai</b>	<b>77</b>
	<b>1 priedas. Atvejo analizė: trijų sluoksnių CNN su vienu papildomu statiniu duomenų sluoksniu</b>	<b>77</b>

2 priedas. Įvairių architektūrų prognozių tikslumo įvertinimo rezultatų suvestinė	80
3 priedas. Atvejo analizė: įvairių CNN architektūrų generuojamų prognozių pavyzdžiai	83
4 priedas. Pirmieji trumpalaikių kritulių prognozių tyrimo, naudojantis Doplerio radaro kritulių radaro duomenis Lietuvoje, rezultatai	88

## Lentelių sąrašas

1	Vidutinė kritulių judėjimo kryptis kiekvieną mėnesį . . . . .	44
2	Klaidų matrica . . . . .	46
3	Trijų sluoksnių CNN modelių su skirtingais papildomais kanalais RMSE palyginimo lentelė . . . . .	57
4	Trijų sluoksnių CNN su įvairiais papildomų kanalų deriniais vidutinės RMSE palyginimas . . . . .	59
5	Įvairių neuroninių tinklų architektūrų apmokymo trukmė ir apmokomų parametrų kiekis . . . . .	61
6	Analizuotų architektūrų RMSE įverčio rezultatai . . . . .	80
7	Analizuotų architektūrų HK įverčio rezultatai . . . . .	81
8	Analizuotų architektūrų FSS įverčio rezultatai . . . . .	82

## Iliustracijų sąrašas

1	Meteorologinio radaro duomenų atvaizdavimas žemėlapiu . . . . .	12
2	Vidutinis metinis kritulių kiekis Lietuvoje ir Lietuvos reljefas . . . . .	14
3	Lietuvos meteorologinių radarų vieta ir padengimas . . . . .	16
4	Vėjo jėginių įtaka meteorologinio radaro rezultatams . . . . .	18
5	Audros judėjimo vektoriai vienos valandos intervale . . . . .	18
6	Santykinis trumpalaikių, vidutinės trukmės ir klasikinių prognozių tikslumo pa- lyginimas . . . . .	19
7	Ranka rašytų skaičių klasifikacija . . . . .	21
8	Pilnai sujungto dirbtinio neuroninio tinklo struktūra . . . . .	22
9	Konvoliucinis neuroninis tinklas ir paveikslėlio atpažinimas . . . . .	22
10	Regresijos metodų ir profesionalių prognozių vidutinės kvadratinės paklaidos palyginimas . . . . .	23
11	Prognozės metodų tikslumo palyginimas . . . . .	26
12	Radaro paveikslėliai - vaizdo įrašo kadrai . . . . .	27
13	Autoenkoderio ir konvoliucinių neuroninių tinklų architektūra žmogaus veiksmo prognozavimui . . . . .	27
14	Panaudos atvejų modelis . . . . .	30
15	Sistemos konteksto diagrama . . . . .	31
16	Statinis sistemos vaizdas . . . . .	32
17	Sistemos esybių ryšių diagrama . . . . .	33
18	Prognozių generavimo serviso bendradarbiavimo diagrama . . . . .	34
19	Duomenų analizės serviso bendradarbiavimo diagrama atnaujinant kritulių prog- nozes . . . . .	35
20	Sistemos diegimo diagrama . . . . .	36
21	Viešai prieinama kritulių radaro informacija . . . . .	38
22	Geografinis duomenų srities padengimas . . . . .	39
23	Apdorotų radaro paveikslėlių seka . . . . .	41
24	Laikotarpis su didžiausiu užfiksuotu kritulių padengimu ir bendru kritulių stiprumu	42
25	Vidutinės mėnesio kritulių charakteristikos . . . . .	43
26	Dažniausia kritulių judėjimo kryptis pagal žemėlapio masės centrą . . . . .	44
27	Vidutinis kiekvieno mėnesio kritulių judėjimo greitis pikseliais per valandą . . . .	45
28	Ankstyvųjų eksperimentų su deterministiniais modeliais rezultatai . . . . .	51
29	3-jų sluoksnių konvoliucinis neuroninis tinklas . . . . .	52
30	4-ių sluoksnių konvoliucinis neuroninis tinklas . . . . .	53
31	Autoenkoderio tipo konvoliucinis neuroninis tinklas . . . . .	54
32	Konvoliucinio LSTM tinklo sluoksnio architektūra . . . . .	55
33	Atsitiktinių reikšmių ir vietovės reljefo kanalas . . . . .	56
34	Visų trijų sluoksnių konvoliucinių neuroninių tinklų RMSE palyginimas . . . . .	57
35	Geriausių kiekvienos rūšies trijų sluoksnių konvoliucinių neuroninių tinklų paly- ginimas . . . . .	58
36	Trijų sluoksnių CNN su įvairiais papildomų kanalų deriniais palyginimas . . . . .	60



37	4-ių sluoksnių CNN prognozių tikslumo pagal įvairias matavimo metrikas palyginimas . . . . .	62
38	Autoenkoderio prognozių tikslumo įverčių pagal įvairias matavimo metrikas palyginimas . . . . .	63
39	Konvoliucinio LSTM tinklo prognozių tikslumo įverčių pagal įvairias matavimo metrikas palyginimas . . . . .	63
40	Geriausi kiekvienos architektūros modeliai pagal RMSE metriką . . . . .	64
41	Geriausi kiekvienos architektūros modeliai pagal HK metriką . . . . .	65
42	Geriausi kiekvienos architektūros modeliai pagal FSS metriką . . . . .	66
43	Trijų sluoksnių CNN generuojamų prognozių palyginimas: 1-as atvejis . . . . .	77
44	Trijų sluoksnių CNN generuojamų prognozių palyginimas: 2-as atvejis . . . . .	78
45	Trijų sluoksnių CNN generuojamų prognozių palyginimas: 3-ias atvejis . . . . .	79
46	Įvairių architektūrų CNN generuojamų prognozių palyginimas: 1-as atvejis . . . . .	83
47	Įvairių architektūrų CNN generuojamų prognozių palyginimas: 2-as atvejis . . . . .	84
48	Įvairių architektūrų CNN generuojamų prognozių palyginimas: 3-ias atvejis . . . . .	85
49	Įvairių architektūrų CNN generuojamų prognozių palyginimas: 4-as atvejis . . . . .	86
50	Įvairių architektūrų CNN generuojamų prognozių palyginimas: 5-as atvejis . . . . .	87

# Santrauka

Nepaisant nuolat augančių superkompiuterių pajėgumų, tradicinės analitinės orų prognozės vis dar nėra laikomos pakankamai tiksliais, o vieno prognozės modelio algoritmo skaičiavimai gali užtrukti net iki poros valandų. Dėl šios priežasties modeliams sunku prognozuoti žaibiškai susidarantiems meteorologiniams reiškiniams, galintiems padaryti nemažai žalos turtui ar net pareikalauti žmonių gyvybių. Išryškėja ir didžiulių energijos resursų suvartojimo problema.

Šio magistro projekto metu kuriami ir pristatomi mašininiai mokymai ir dirbtiniais neuroniniais tinklais paremti kritulių prognozavimo algoritmai, veikiantys Lietuvos teritorijoje ir leidžiantys greitai gauti trumpalaikių kritulių prognozių rezultatus. Siekiant sukurti kokybiškus orų prognozių modelius ir pasisemti rezultatų analizei reikalingos patirties, atliekama meteorologijos procesų, kritulių radaro ypatumų ir trūkumų analizė, pristatomos pagrindinės dirbtinių neuroninių tinklų sąvokos, architektūros ir taikymai dalykinėje srityje.

Sukurtų algoritmų rezultatų tikslumą tinklapyje [kadalis.lt](http://kadalis.lt) galima išbandyti praktiškai: peržiūrėti sugeneruotų prognozių žemėlapius ar skaitines kritulių prognozes savo mieste ir palyginti skirtingų algoritmų sugeneruotus kritulių spėjimus tarpusavyje.

Algoritmų apmokymui, validavimui ir testavimui surinkta daugiau nei 70 000 meteorologinio Doplerio kritulių radarų žemėlapių, aprėpiančių 2017–2020 metų laikotarpį. Su apmokymui skirtais duomenimis atliekama kritulių dažnio, judėjimo krypties ir greičio tendencijų apžvalga. Aprašomi ir tyrime naudojami prognozių tikslumo matavimo algoritmai: šaknies iš vidutinės kvadratinės paklaidos (RMSE), Hansen'o-Kuiper'io (HK) ir dalinio tikslumo (FSS) įverčiai, jų privalumai bei trūkumai.

Susipažinus su sukurtų mašininiai mokymai besiremiančių algoritmų architektūra, šių modelių prognozių rezultatai trijuose eksperimentuose lyginami su faktinėmis orų sąlygomis: pirmajame eksperimente tiriama papildomų statinių įvesties sluoksnių įtaka 3-jų sluoksnių konvoliucinio neuroninio tinklo gaunamiems rezultatams, antrajame – šių statinių sluoksnių kiekio įtaka generuojamų prognozių tikslumui, trečiajame – atliekama išsami skirtingų architektūrų neuroninių tinklų generuojamų prognozių tikslumo analizė su pilna testavimo duomenų aibe (daugiau nei 10 000 žemėlapių egzempliorių).

Visi sukurti algoritmai pagal visas tikslumo matavimo metrikas smarkiai aplenkė spėjimo, kad pradinės sąlygos ateityje nepasikeis, tikslumo įvertį, o geriausius rezultatus parodė santykinai nesudėtingas keturių sluoksnių konvoliucinio neuroninio tinklo algoritmas. Rezultatai parodė ir papildomų statinių duomenų sluoksnių konvoliucinio neuroninio tinklo įvestyje naudą: papildomi atsitiktiniai ir tiriamos vietovės reljefą atitinkantys duomenys pagerino analizuotų trijų sluoksnių konvoliucinių neuroninių tinklų pasiekiamą tikslumą. Gauti rezultatai atspindi naujų skaičiavimo metodų paieškos svarbą srityse, kuriose jau senokai nusistovėję tradiciniai matematiniai rezultatų gavimo metodai: jei mašininiai mokymai algoritmai tobulės ir toliau, orų prognozės bus generuojamos greičiau, sueikvojant mažiau energijos. Magistrinis projektas akcentuoja meteorologijos ir dirbtinio intelekto mokslų tarpusavio bendradarbiavimo svarbą.

**Raktažodžiai: meteorologija, krituliai, orų prognozavimas, konvoliuciniai neuroniniai tinklai**

# Summary

Although the power of supercomputers is constantly increasing, the traditional analytical approaches to weather forecasting are still not considered to be accurate by the general public. In addition to this, the duration of a single forecast computation sometimes can take up to 2 hours. This means that weather prediction models sometimes fail to react to rapid and dangerous changes in weather conditions, which may result in the destruction of material possessions or cost human lives.

This master's thesis describes the creation and analysis of precipitation forecast algorithms, which are based on machine learning and artificial neural networks. These algorithms are designed for fast short-term precipitation forecasting over the territory of Lithuania and rely on Doppler's weather radar images. To create accurate machine learning models and gain more experience in result analysis, related work, which concerns meteorology, precipitation movement, weather radar technology, machine learning, and its applications to weather forecasting is presented.

The results of the analyzed algorithms can be viewed using interactive [kadalis.lt](http://kadalis.lt) webpage. This platform allows users to view generated precipitation forecast maps and informs them about predicted conditions in their city. The webpage also gives an ability to compare forecast maps generated by the different machine learning algorithms.

More than 70 000 different weather radar images were obtained to be used for the training, validation, and testing of the created algorithms. The exploration results of the training data are also presented and consist of precipitation frequency, movement direction, and speed analysis, which helped in choosing some of the model's hyperparameters. Additionally, forecast accuracy measuring techniques (Root-Mean-Squared Error, RMSE; Hanssen-Kuiper score, HK; and Fraction-Skill Score, FSS) are described.

After a brief description of an architecture of the created machine learning algorithms, these models are evaluated during three experiments: the first experiment analyses an impact of additional static input data layers on the results of a 3-layer convolutional neural network, the second explores the influence of a different number of these layers to the accuracy of this same network, and the third one describes an analysis of multiple different neural network architectures for precipitation prediction, using more than 10 000 validation examples.

All of the created algorithms have surpassed the accuracy of a simple persistency model, and the convolutional neural network with 4 layers showed the best results. A positive impact of the additional static layers (random and elevation data) in the model input was also observed in the convolutional neural network with 3 layers. Obtained results show the importance of a search for new algorithms in areas that already have stable traditional mathematical methods: if the technological progress in machine learning algorithms continues, weather forecasts will be generated faster and will use less amount of energy. This master's thesis highlights the importance of interdisciplinary dialog between artificial intelligence and meteorology scientist communities.

**Keywords:** meteorology, precipitation, weather forecasting, convolutional neural networks

# 1 Įžanga

## 1.1 Projekto naujumas ir aktualumas

Magistro projekto metu tiriamas dirbtiniais neuroniniais tinklais besiremiančių algoritmų panaudojimas kritulių prognozavimui pagal meteorologinio radaro žemėlapius. Dirbtiniai ir gilieji neuroniniai tinklai yra nuolat tobulėjantis mokslas, pasiekiantis vis geresnius rezultatus įvairiausiose taikymo srityse. Tyrimo metu siekiama išanalizuoti mašininio mokymosi algoritmų tikslumą srityje, kurioje jau seniai nusistovėję tradiciniai matematiniai prognozės modeliai.

Pagal viešai prieinamą informaciją, šis tyrimas yra pirmasis, kuriame aprašomas mašininio mokymosi paremtų kritulių prognozavimo algoritmų tikslumas, kai modelių apmokymui ir validavimui naudojami Lietuvos teritorijoje užfiksuoti duomenys.

## 1.2 Tikslas ir uždaviniai

Šio baigiamojo magistro projekto tikslas – sukurti mašininio mokymosi metodais paremtus kritulių prognozavimo algoritmus ir išmatuoti jų tikslumą.

Tikslui pasiekti keliami šie uždaviniai:

1. Apibrėžti atliekamo tyrimo ir kuriamų kritulių prognozės algoritmų praktinę naudą gyventojams, mokslininkams, bei įvairioms organizacijoms, aprašant trumpalaikių orų prognozių pranašumus prieš tradicinius metodus. Pristatyti mašininio mokymosi paremtų modelių perspektyvumą ir inovatyvumą.
2. Išanalizuoti esamą literatūrą bei mokslinius straipsnius ir pristatyti sąvokas, idėjas bei procesus, kurie gali būti naudingi tyrimo ir eksperimento metu. Kritulių susidarymo ir judėjimo procesų aprašymu siekiama bent minimaliai susipažinti su meteorologijos mokslu, iš kurio būtų galima pasisemti bazinių žinių apie fizikinius procesus, veikiančius kritulių dinamiką. Verta susipažinti ir su tradiciniais metodais, besiremiančiais kritulių judėjimo ekstrapoliacija ir pagrindiniu šio darbo duomenų šaltiniu – meteorologiniu Doplerio kritulių radaru, jo privalumais bei trūkumais.
3. Pristatyti esminius darbe naudojamų dirbtinių neuroninių tinklų (angl. *Artificial Neural Networks, ANN*) architektūrų aspektus ir jų taikymus meteorologinėse prognozėse. Geriausios neuroninio tinklo architektūros ir jos hiperparametrų pasirinkimas konkrečiai sprendžiamai problemai dažnai panašesnis į didelės patirties reikalaujantį meną, o ne į griežtinus taisyklėmis paremtą mokslą. Dėl šios priežasties, renkantis neuroninio tinklo dizainą, būtina susipažinti su neuroninių tinklų architektūromis, pasiekusiomis geriausius rezultatus meteorologijos ar susijusiose taikymo srityse.
4. Trumpai aprašyti sukurtą programinę įrangą – internetinę platformą „kadalis.lt“, skirtą įvairių algoritmų prognozių rezultatų atvaizdavimui realiu laiku Lietuvos teritorijoje. Platforma siūlo interaktyvią nuolat atnaujinamų prognozių peržiūrą: vizualus rezultatų, naudojančių realius duomenis, pateikimas leidžia geriau įsivaizduoti įvairių algoritmų tikslumą, o prognozes galima iškart panaudoti praktiškai.

5. Charakterizuoti naudojamus duomenis: jų struktūrą, specifiką, datos intervalą ir sprendimus efektyviam didelio jų kiekio saugojimui; pasirinkti geriausias metodus prognozių tikslumo matavimui, lyginant algoritmų gautus rezultatus su faktiniais duomenimis. Norint geriau suprasti apmokymui skirtus duomenis, būtina juos išsamiai išanalizuoti ir surasti galimas kritulių susidarymo ir judėjimo tendencijas.
6. Apmokyti skirtingus dirbtinių neuroninių tinklų algoritmus ir įvertinti jų tikslumą naudojant apibrėžtus metodus ir fiksuotą testavimo duomenų rinkinį. Gautus tikslumo rezultatus palyginti tarpusavyje ir nustatyti algoritmą(-us), geriausiai tinkančius trumpalaikiam kritulių prognozavimui. Algoritmų rezultatus atvaizduoti sugeneruotų kritulių žemėlapiams, iliustruojančiais pasirinktus atvejus.

### 1.3 Dokumento struktūra

Magistrinis projektas pradedamas analitine dalimi (žr. 2 skyrių) – pristatomi pagrindiniai kritulių susidarymo ir judėjimo principai, kritulių stebėjimo įrankiai ir jų trūkumai. Susipažįstama su trumpalaikėmis kritulių prognozavimo metodologijomis, pristatomos esminės mašininio mokymosi sąvokos ir šios mokslo srities taikymas kritulių prognozavimo bei vaizdo įrašų prognozavimo srityse.

Projektinėje dalyje (žr. 3 skyrių) trumpai pristatoma sukurta interaktyvi programinė įranga, leidžianti peržiūrėti ir palyginti skirtingų kritulių prognozės algoritmų rezultatus: pateikiama galimybė tyrimo rezultatus įvertinti ne tik teoriškai, bet ir praktiškai.

Tyrimo dalyje (žr. 4 skyrių) pristatoma išsami per dvejus metus surinktų Doplerio kritulių radaro sugeneruotų kritulių žemėlapių analizė ir pastebėtos tendencijos. Susipažįstama ir su eksperimentuose naudojamais kritulių prognozių tikslumo matavimo būdais ir jų taikymo specifiška.

Eksperimento dalyje (žr. 5 skyrių) pateikiami algoritmų sugeneruotų prognozių tikslumo tyrimo rezultatai ir tarpusavio palyginimas.

Darbo pabaigoje pateikiamos išvados (žr. 6 skyrių), akronimų sąrašas, glosarijus ir naudotų literatūros šaltinių sąrašas.

Prieduose galima rasti algoritmų sugeneruotų prognozių pavyzdžius, detalias eksperimentų rezultatų lenteles ir konferencijoje pristatytą straipsnį apie trumpalaikį kritulių prognozavimą Lietuvoje.

## 2 Analitinė dalis

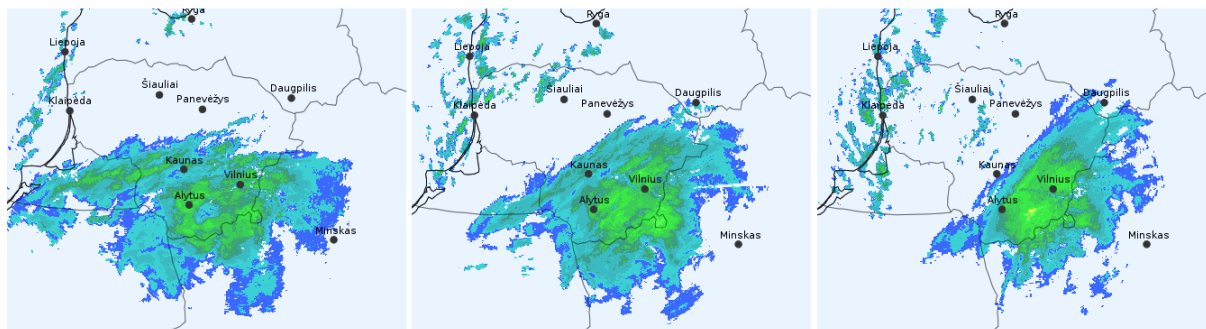
Šiame skyriuje pristatoma medžiaga yra magistrinio projekto tyrimo teorinis pagrindas. Pristatomos meteorologinės prognozės, fizikiniai kritulių susidarymo ir judėjimo principai, Doplerio kritulių radaras ir trumpalaikės kritulių prognozavimo technologijos: tradiciniai bei mašininiai mokymusi paremti metodai.

### 2.1 Analitinės dalies įvadas

Šiais laikais orų prognozių sudarymo mokslas remiasi sudėtingais fizikiniais modeliais, kurie naudoja duomenis iš dirbtinių žemės palydovų bei meteorologinių stočių ir remiasi painiomis fluidų dinamikos lygtimis. Orų modeliavimas atliekamas galingais superkompiuteriais, kurie kainuoja milijonus dolerių ir savo dydžiu primena pačius pirmuosius kompiuterius, vos telpančius į didelį sandėlį. Jų veikimui ir aušinimui reikalingos didžiulės elektros energijos sąnaudos.

Nepaisant sunkiai įsivaizduojamos šių milžinų galios, orų prognozės (ypač ilgesniam nei savaitės laikotarpiui) vis dar nėra tikslios, o skaičiavimai trunka iki poros valandų. Tai susiję su orų prognozių nedeterminiškumu: net ir turint kone idealų meteorologinių stočių ir dirbtinių žemės palydovų tinklą, neįmanoma žinoti visų orus veikiančių pradinių sąlygų. Diferencialinių fluidų dinamikos lygčių neįmanoma išspręsti analitiškai, tad orų prognozių modeliai naudoja įvairias euristicas, kurių netikslumai laikui bėgant lemia vis didesnę prognozių nukrypimą nuo realių orų sąlygų.

Net ir trumpalaikės orų prognozės nėra tobulos: tai ypač pasireiškia prognozuojant staigias vasaros audras, kurios priklauso nuo vietinių gamtos sąlygų. Šių sąlygų pasauliniai orų prognozių modeliai dažnai neįvertina. Be to, dėl ilgos prognozių apskaičiavimo trukmės, prognozės rezultatai gali tapti nebeaktualūs: svarbus meteorologinis reiškinys ar stichinė nelaimė gali įvykti kompiuteriams dar net nepabaigus skaičiavimų.



1 pav. Meteorologinio radaro duomenų atvaizdavimas žemėlapiu<sup>1</sup>

Trumpalaikės orų prognozės gali būti sudaromos ir ekstrapoliuojant faktinius duomenis apie meteorologines sąlygas, tokius kaip meteorologinio radaro žemėlapiai (žr. 1 pav.). Kadangi tokie modeliai atsižvelgia į vietines orų sąlygas, jų tikslumas konkrečiam regionui dažnai būna geresnis, nei sudėtingų pasaulinių modelių. Tiesa, laikui bėgant šių modelių prognozavimo kokybė sparčiai mažėja: prognozės tokiu būdu gali būti sudaromos vos keletui valandų.

<sup>1</sup>Žemėlapiai pateikiami 2 val. intervalais, nuo 2017-10-29 11:15 iki 15:15. Šaltinis: LHMT, Radarų informacija, <http://www.meteo.lt/lt/radaru-informacija>

Įvairių objektų ir reiškinių, kuriems sunku sukurti tikslius algoritmus, klasifikavimui ar regresija paremtai prognozei gali būti naudojami mašininio mokymosi algoritmai. Pavyzdžiui, kompiuteriui pateikus didelę meteorologinio radaro paveikslėlių aibę ir pritaikius tinkamą apmokymo algoritmą, mašina gali „išmokti“, kaip juda krituliai ir prognozuoti jų kelią netolimoje ateityje.

Algoritmo apmokymas gali trukti pakankamai ilgai, tačiau kokybiškai jį atlikus, rezultatai gali būti gaunami kur kas greičiau, nei naudojant sudėtingus matematinis orų modelius. Toks ar panašus metodas gali būti pritaikomas greitai gaunamoms trumpalaikėms orų prognozėms, pasižyminčioms ne ką mažesniu tikslumu nei tradiciniai matematiniai modeliai.

Trumpalaikės orų prognozės ypač svarbios stichinių reiškinių, tokių kaip stiprios audros ar liūtys, prognozavimui. Šie vos per kelias valandas susidarantys reiškiniai sukelia daug problemų: staigūs potvyniai, nuošliaužos, stichinės liūtys gali ne tik padaryti daug materialinės žalos, tačiau ir kainuoti žmonių gyvybes.

## 2.2 Kritulių susidarymo ir judėjimo principai

Norint sukurti gerą mašininiam mokymuisi skirtą modelį, reikia išsiaiškinti, kokie fizikiniai procesai sąlygoja kritulių susidarymą ir judėjimą, bei kiek šiems procesams įtakos turi įvairios aplinkos sąlygos. Šiame poskyryje trumpai apžvelgiami kritulių tipai, jų susidarymas ir kritimo priežastys.

### 2.2.1 Kritulių susidarymas

Krituliai – tai iš vandens sudarytos dalelės, susiformuojančios atmosferoje ir krentančios ant žemės dėl gravitacijos poveikio. Pagal pavidalą jos skirstomos į tris pagrindinius tipus: vandens lašelius, krušą bei sniegą.

Vandens lašeliai atsiranda smulkioms vandens dalelėms susiduriant ir jungiantis į vis didesnes. Jens'as Eggers'as et al. šį reiškinį – vandens lašelių susijungimą, suaugimą (angl. *Coalescence*) – apibrėžia taip: kai du lietaus lašeliai su spinduliu  $R$  susiliečia, paviršiaus įtempimo jėga sukelia į centrą juos abu traukiantį judėjimą, kuris sujungia juos į didesnę lašelį, turintį mažesnę paviršiaus plotą [1].

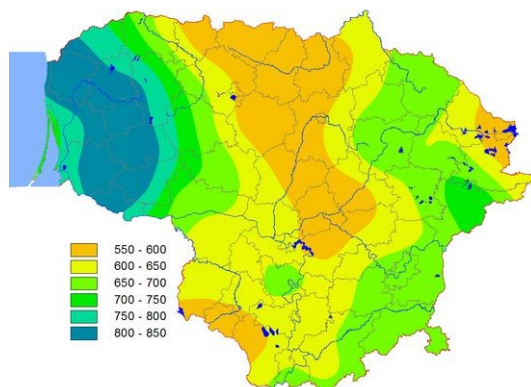
Šis reiškinys, jungiantis vis didesniems lašeliams, kartojasi tol, kol vieno lašelio svoris pasidaro per didelis, kad išsilaikytų ore ir šis ima kristi žemyn. Vidutinėse Žemės platumose tokiu būdu susiformavę krituliai paprastai sudaro didžiausią kritulių dalį.

Kritulių susidarymą ir iškritimą lemia trys pagrindinės priežastys:

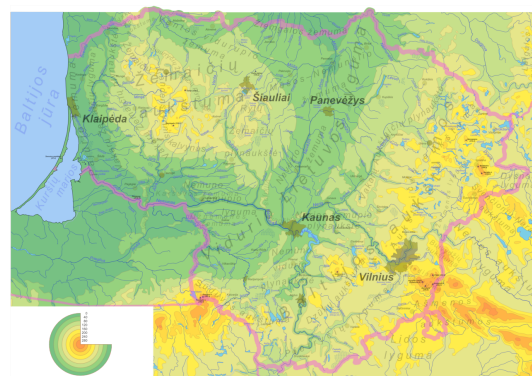
- atmosferos frontų judėjimas;
- konvekcija;
- Žemės paviršiaus reljefas.

Atmosferos frontai susidaro dėl skirtingų temperatūrų oro masių judėjimo. Krituliai susiformuoja frontų susikirtimo vietose ir priklauso nuo trijų fronto tipų: šaltojo, šiltojo ir okliuzijos.

R. J. Barry'is ir R. J. Correy'is šiltąjį atmosferos frontą ir jo sąlygojamus kritulius apibūdina taip: šiltasis frontas atitinka šiltojo sektoriaus kraštą. Krituliai paprastai būna lengvas lietus



(a) Vidutinis metinis kritulių kiekis Lietuvoje (mm)



(b) Aukštis virš jūros lygio Lietuvos teritorijoje

**2 pav.** Vidutinis metinis kritulių kiekis Lietuvoje (a) ir Lietuvos reljefas (b). Nereikia didelių kalnų, kad pasireikštų lietaus šešėlio efektas<sup>2</sup>

ar dulksna, susiformavusi dėl lietaus lašelių tarpusavio jungimosi. Šaltojo fronto judėjimo zonoje susidarantys krituliai aprašomi kaip trumpos, stiprios liūtys, kartais lydimos griaustinio. Okliuzija susidaro šaltajam frontui aplenkus šiltąjį: šaltasis frontas juda greičiau nei šiltasis ir galiausiai jį pasiveja, kas sąlygoja okliuzijos susidarymą. Okliuzija sąlygoja įvairių tipų kritulius [2].

Kita kritulių priežastis – konvekcija. Ji yra svarbi vandens ciklo dalis ir vyksta, kai saulė įkaitina įvairius vandens šaltinius Žemėje: jie garuoja, vandens lašeliai suformuoja debesis, kurie, neilgai trukus, sukelia įvairių tipų kritulius. Konveciniai lietūs paprastai trunka trumpai, apima nedidelę teritoriją, tačiau gali būti labai stiprūs. Beveik visi krituliai tropikuose yra konvecinio pobūdžio [3].

Šio tipo krituliai gali būti iššūkis mašininiu mokymusi paremtam kritulių judėjimo modeliui: konveciniai krituliai dažnai susiformuoja ir iškrenta toje pačioje vietovėje. Kritulių prognozavimo modelis gali nespėti įvertinti tokių dinamiškų kritulių judėjimo greičio ir krypties. Neturint papildomos informacijos sunku prognozuoti ir tokių kritulių susidarymą. Siekiant patobulinti modelį, reikėtų papildomai įvertinti ir temperatūrą bei slėgį, o detalią šių parametrų istoriją, apimančią visą Lietuvos teritoriją, gauti sudėtinga.

Didelę įtaką kritulių susidarymui turi Žemės reljefas. G. H. Roe'us straipsnyje apibrėžia klasikinį orografinių kritulių vaizdą: kalnų grandinė užstoja orų srautus, kurie atneša kritulius. Vienoje kalnų grandinės pusėje (vertinant klimatologinį vidurkį) vyrauja drėgni orai, o kitoje – sausi. Šis efektas dar žinomas kaip „lietaus šešėlis“.

Jei regione orų masės linkusios atkelti tik iš tam tikros pusės, orografinis pakilimas gali būti esminis kritulių (arba jų nebuvimo) šaltinis: pavyzdžiui, Lietuvoje, kur orų masės atkeliauja nuo Baltijos jūros, vakarinėje Žemaičių aukštumos pusėje iškrenta daugiau kritulių nei rytinėje (žr. 2 pav.). Net ir esant pakankamai mažiems kalnams (Lietuvoje iki 300 m), reljefo įtaka vidutiniam kritulių kiekiui yra pakankamai didelė (skirtumas iki 300 mm kritulių per metus). Kitose pasaulio vietose, kur kalnai aukštesni, šis skirtumas yra dar didesnis: kai vienoje kalnų pusėje dėl gausių kritulių vyrauja derlingas dirvožemis, kitoje gali plytėti sausa

<sup>2</sup>Šaltiniai: Lietuvos klimatas – Lietuvos hidrometeorologijos tarnyba (a) ir *Wikipedia* (b)



ir negyva dykuma.

Akivaizdu, kad reljefas turi didelę įtaką krituliams. Dėl šios priežasties mašininio mokymosi modelio tikslumas galėtų būti padidintas jį papildant duomenimis apie tiriamos vietovės reljefo ypatumus.

Lietuva yra vidutinių platumų klimato zonoje, tad dažniausiai kritulius čia atneša atmosferos frontai bei jūrinės oro masės. Konvekciniai krituliai yra retesni: jie dažniausiai susidaro karštomis ir saulėtomis vasaros dienomis, kai saulės šilumos užtenka stiprių, bet neilgai trunkančių audrų susiformavimui. Kaip pastebima šiame poskyryje, **orografiniai krituliai** Lietuvai turi taip pat nemažai įtakos.

Skirmantė Strolytė ir Egidijus Rimkus savo straipsnyje nusako Lietuvoje istoriškai vyraujančią vėjo kryptį (kuri gali bendrais bruožais nusakyti iš kurios pusės dažniausiai atslenka oro masės, taip pat ir krituliai): „1971–2010 m. beveik visoje Lietuvos teritorijoje vyravo vakarinių ir pietinių rumbų vėjai, nors Klaipėdoje gana dažnai fiksuoti ir rytinių rumbų vėjai. Visose meteorologijos stotyse rečiausiai pasitaikė šiaurinių rumbų, o Utenoje ir Kaune – ir rytų rumbų vėjai.“ [4]

Kadangi vėjo kryptis dažnai gali atitikti kritulių judėjimo kryptį, ši informacija gali būti naudinga vertinant turimus duomenis ir algoritmų tikslumą, pavyzdžiui, užtikrinant, kad kritulių prognozių algoritmai neįsiminė dažniausiai pasitaikančios kritulių judėjimo krypties.

## 2.3 Meteorologinis radaras

Pagrindinis meteorologų įrankis stebint kritulius ir jų judėjimą yra Doplerio kritulių radaras. Tai – pagrindinis šiame darbe analizuojamų duomenų šaltinis, tad verta išsiaiškinti šio prietaiso veikimo principus, stipriąsias puses bei apribojimus.

### 2.3.1 Doplerio radaro technologija

Meteorologinis radaras – tai mikrobangų radaras, skirtas kritulių stiprumo bei judėjimo krypties stebėjimui realiuoju laiku. R. J. Doviak'as ir D. S. Zrnici'us savo knygoje apibrėžia tokį mikrobangų pranašumą prieš kitas elektromagnetinio spektro dedamąsias: mikrobangų gebėjimas prasiskverbti pro debesis ir lietu leidu meteorologiniam radarui tapti svarbiausiu prietaisu nuotoliniame atmosferos stebėjime. Ir nors matomos šviesos ir infraraudonųjų spindulių kameros palydovuose gali aptikti ir stebėti audrų judėjimą, šių kamerų juntama radiacija negali prasiskverbti pro audros debesų šydą, taip kaip prasiskverbia mikrobangų radaras [5].

Meteorologinis radaras krituliams aptikti norima kryptimi išsiunčia mikrobangų pulsą. Šios bangos, atsispindėjusios nuo kokio nors objekto (taip pat ir įvairaus tipo kritulių), grįžta atgal į radaro imtuvą. Pagal laiką, praėjusį tarp bangų išsiuntimo ir grįžimo, galima apskaičiuoti objekto atstumą iki radaro, o pagal atspindėtų bangų stiprumą – kritulių intensyvumą.

Pasinaudodami Doplerio efektu (iš kurio ir kilo tokio tipo radaro pavadinimas), modernūs kritulių radarai gali apskaičiuoti ne tik kritulių stiprumą ar jų poziciją, tačiau ir judėjimo kryptį. Morgan'as Bettex'as savo straipsnyje Doplerio efektą aprašo taip: Doplerio efektas, arba Doplerio poslinkis, aprašo bet kokio tipo garso ar šviesos bangų dažnio pokytį, sukeliama judančio šaltinio stebėtojo atžvilgiu.

Kai objektas juda link stebėtojo, jo skleidžiamų bangų dažnis padidėja, o judant į priešingą pusę – sumažėja. Meteorologinis radaras pagal priimamų bangų dažnio pokytį gali apskaičiuoti, į kurią pusę juda stebimi krituliai. Dažnio pokytis yra proporcingas objekto judėjimo greičiui, tad įmanoma sužinoti ne tik kritulių judėjimo kryptį, tačiau ir jų poslinkio greitį.

Svarbu pastebėti tai, kad nors šie duomenys ir yra prieinami žmonėms, dirbantiems su Doplerio radaru ir analizuojantiems jo neapdorotus duomenis, viešam naudojimui Lietuvos hidrometeorologijos tarnyba pateikia tik galutinius kritulių žemėlapius. Akivaizdu, kad pilni, neapdoroti meteorologinio radaro duomenys turėtų potencialą žymiai pagerinti šiame darbe generuojamų prognozių tikslumą. Tačiau tokių duomenų trūkumą galima vertinti ir kaip papildomą iššūkį: mašininio mokymosi pagalba iš ribotos informacijos galima netiesiogiai išskaičiuoti trūkstamus kintamuosius.

### 2.3.2 Meteorologinis radaras Lietuvoje

Lietuvos teritoriją stebi du meteorologiniai radarai – Laukuvoje (nuo 2010 m.) ir Trakų Vokėje (nuo 2014 m.)<sup>3</sup>. Radarai kartu padengia visą Lietuvos teritoriją (žr. 3 pav.).

Oficialiame LHMT tinklapyje skelbiama, kad šie abu įrenginiai yra dvigubos poliarizacijos Doplerio meteorologiniai radarai. Svarbiausia dvigubos poliarizacijos radaro ypatybė apibūdinama kaip skirtingų kritulių tipų nustatymas. Vasarą radarai geba rinkti duomenis 150 km spinduliu; žiemą – 100 km.

Radaru surinkta informacija pateikiama Lietuvos hidrometeorologijos tarnybos tinklapyje žemėlapiu.



**3 pav.** Lietuvos meteorologinių radarų vieta ir padengimas. Šaltinis: Lietuvos hidrometeorologijos tarnyba

<sup>3</sup>Deja, nuo 2019 m. pastarasis radaras nebeveikia ir nebeteikia duomenų. Gedimas riboja ir duomenų bei rezultatų kokybę šiame projekte, nes joks radaras patikimai nedengia Vilniaus ir jo apylinkių

### 2.3.3 Radaro ribotumas

Atstumas, kuriuo radaras gali patikimai aptikti kritulius, yra apribotas Žemės išlinkimo: dėl šios priežasties tradiciniai meteorologiniai radarai negali aprėpti didesnio nei 100–150 km spindulio. Be to, daugelio radarų konstrukcija neleidžia jų antenai pasisukti statmenai į viršų, tad radaras negali aptikti kritulių, esančių tiesiai virš jo.

Šias problemas galima iš dalies išspręsti aukšto dažnio bangų radaru, kuris geba aptikti kritulius, esančius už horizonto. Tai pasiekama dėl pažeme sklindančių bangų difrakcijos ar į viršų sklindančių bangų atspindžio nuo jonosferos. Ši aukšto dažnio bangų savybė leidžia padidinti kritulių radaro veikimo atstumą iki 200–400 km (pažeme sklindančių bangų atveju) ar 1 000–4 000 km (atspindžio nuo jonosferos atveju) [6].

Žemės išlinkimas nėra vienintelis radaro veikimą apribojantis veiksnys. Ne visos nuo kritulių atspindėtos bangos grįžta atgal į radarą, dėl ko gali sumažėti kritulių aptikimo tikslumas

Žiemą kartu su lietumi gali pasirodyti ir kiti kritulių tipai: sniegas, šlapdriba, lijundra. Pats radaras be papildomų duomenų negali diferencijuoti tarp skirtingų kritulių tipų<sup>4</sup>. Atrodo, užtektų tiesiog apibrėžti, kad jei oro temperatūra yra neigiama – tai bus sniegas, o jei teigiama – lietus. Nors tokiu būdu būtų galima teisingai klasifikuoti daugumą kritulių atveju, dėl sudėtingų atmosferoje vykstančių procesų ir temperatūrų skirtumų įvairiame aukštyje, tai tikrai nebūtų pakankamai tikslu – ypač kai temperatūra svyruoja apie 0°C.

Dar viena problema, su kuria susiduria meteorologinis radaras, yra įvairūs antžeminiai artefaktai, kurie gali būti palaikyti krituliais, nors iš tikrųjų tokiais nėra: tai gali būti Žemės paviršiaus ypatumai (kalvos, kalnai) ar žmogaus sukurti objektai (kaminai, antenos, kiti aukšti statiniai).

Naudojantis Doplerio efektu tokius statiškus objektus galima nesunkiai atfiltruoti: jų atspindimų bangų dažnis nesikeis, tad bus galima daryti išvadą, kad šie objektai stovi vietoje, kai tuo tarpu krituliai niekada nebus statiški.

Deja, egzistuoja ir judančių artefaktų, kuriuos nuo kritulių atskirti yra daug sunkiau. Vienas iš pagrindinių meteorologinio radaro „priešų“ yra vėjo jėgainių parkai. Dėl besisukančių jėgainių menčių judėjimo meteorologinis radaras negali atskirti jų nuo tikrų kritulių.

R. J. Vogt'as et al. savo straipsnyje apie meteorologinių radaro ir vėjo jėgainių tarpusavio sugyvenamumą išskiria tokią problemą, kai kritulių zonos sutampa su vėjo jėgainių plotais: silpnas lietus gali būti supainiojamas su stipriomis audromis, o lietaus zona gali būti klaidingai atvaizduojama kaip daug didesnė, nei pats vėjo jėgainių parkas (žr. 4 pav.) [7].

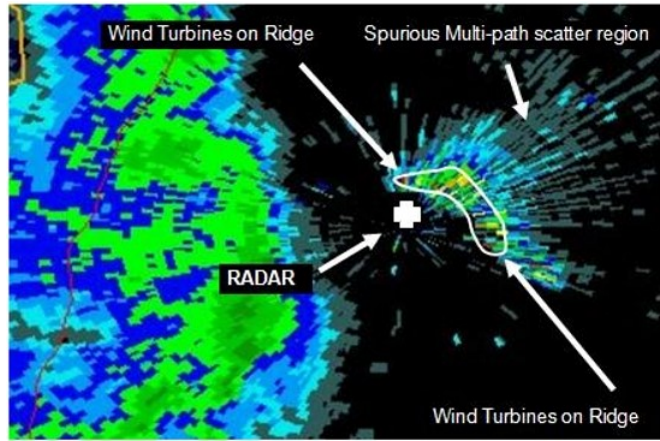
Radaro rodmenims įtakos gali turėti net kasmetinė paukščių migracija [8].

Taigi, galima išskirti pagrindinius kritulių radaro apribojimus, į kuriuos gali tekti atsižvelgti kuriant mašininio mokymosi modelius ar vertinant gautus rezultatus:

- veikimo atstumo ribotumas;
- skirtingų kritulių tipų neskyrimas;
- antžeminiai artefaktai.

---

<sup>4</sup>Bent jau tokie duomenys Lietuvoje nėra pateikiami



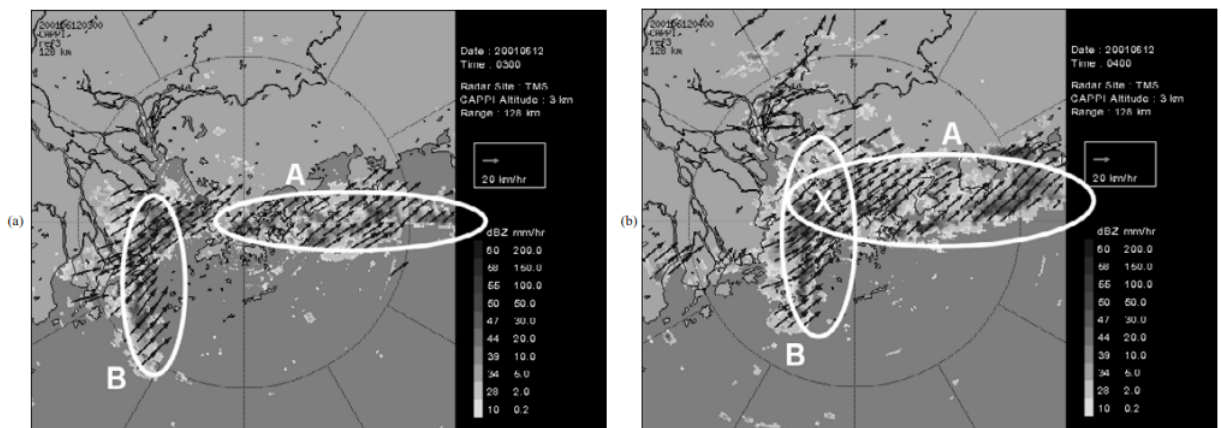
4 pav. Vėjo jėginių įtaka meteorologinio radaro rezultatams. Šaltinis: Amerikos Nacionalinė vandenynų ir atmosferos administracija, NOAA

## 2.4 Trumpalaikės prognozės

Trumpalaikės orų prognozės (angl. *Nowcasting*) apibrėžiamos kaip prognozės laikotarpiui trumpesniam nei keletas valandų. Nuo paprastų vidutinės trukmės prognozių ši sritis skiriasi naudojamais duomenų šaltiniais bei prognozės metodais. J. W. Wilson'as et al. savo straipsnyje išreiškia tokio tipo prognozių svarbą staigiems orų pokyčiams jautrioms sritims: komercinei ir bendrajai aviacijai, lauko sporto renginiams, statybų industrijai, elektros energijos perdavimo įrenginiams ir antžeminiam transportui [9].

Trumpalaikės prognozės gerai tinka greitai susidarančių oro reiškinių, tokių kaip stiprūs konvekciniai lietūs ar audros su žaibais, spėjimui. Tradiciniai prognozės metodai šiuo atveju ne visada yra tinkami, nes dažnai tokios audros susiformuoja ir išnyksta greičiau nei sudėtingi orų prognozės modeliai pateikia savo rezultatus.

Pagrindiniai konvekcinių audrų aptikimo įrankiai yra meteorologiniai radarai, žaibų detektoriai ir palydovinės nuotraukos. Trumpalaikis konvekcinių audrų judėjimo prognozavimas istoriškai daugiausia buvo paremtas radaro atspindžio aidų ekstrapoliacija [9]



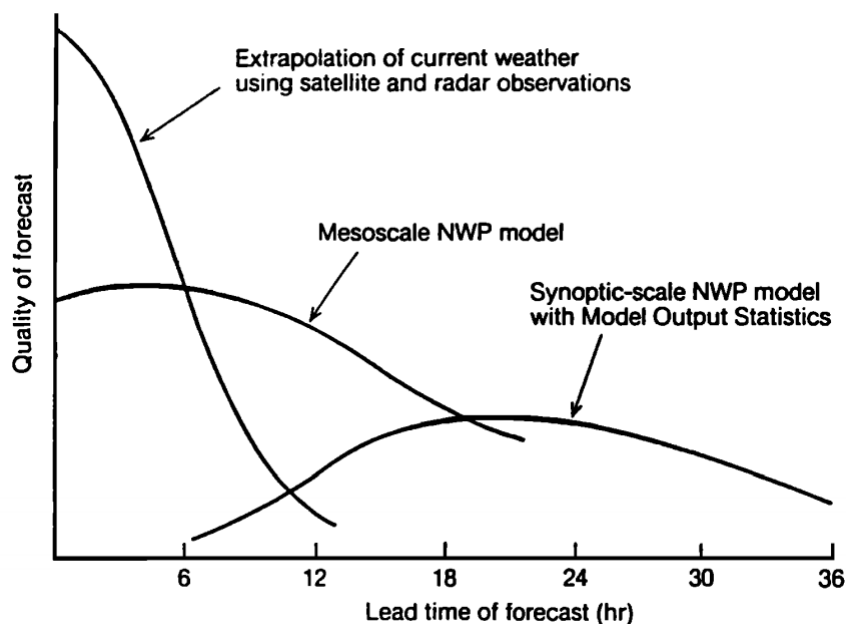
5 pav. Audros judėjimo vektoriai vienos valandos intervale. Šaltinis: [10]

Meteorologinio radaro duomenys dažnai naudojami trumpalaikėms kritulių prognozėms sudaryti. Remiantis šiuo metodu, bandoma iš kelių vienas po kito laike išsidėsčiusių radaro

duomenų rinkinių ekstrapoliuoti kritulių poziciją ir stiprumą keletui valandų į ateitį. L. Li, W. Schmid'as ir J. Joss'as savo straipsnyje išskiria du ekstrapoliacijos būdus: Pirmajame bandoma rasti geriausią atitikimą tarp dviejų tam tikru laiko intervalu gautų radaro duomenų. Kaip duomenų atitikimo kriterijus naudojamas koreliacijos koeficientas. Didžiausias koreliacijos koeficientas atitinka kritulių judėjimo vektorius, kas leidžia ekstrapoliuoti šiuos duomenis į ateitį. Antrasis metodas seka ir prognozuoja regionus, masės centrus ir kitus individualių konvekcinių audrų ar ląstelių parametrus [11].

Šie metodai nėra tik teoriniai modeliai – jie plačiai naudojami sudarant orų prognozes, kai labai svarbus greitas reagavimas į susidariusią situaciją. P. W. Li, Edwin'as ST ir Lai savo straipsnyje aprašo abiejų aukščiau įvardytų metodų panaudojimą vidutinės trukmės orų prognozėms Hong Konge (žr. 5 pav.). Minimi ir šio metodo trūkumai: pats pagrindinis įvardijamas kaip audros judėjimo vektorių prognozavimas, judant ir jungiantis kelioms skirtingoms audrų sistemoms [10].

K. A. Browning'as ir C. G. Collier'as savo straipsnyje aprašo dabartinių oro sąlygų ekstrapoliacijos efektyvumą prieš tradicinius metodus (žr. 6 pav.), tačiau teigia, kad audringas oras, sukeltas individualių konvekcinių sistemų, gyvuoja trumpai ir vargu ar vien tik tiesinė ekstrapoliacija gali pateikti geras prognozes daugiau negu vienai valandai į ateitį [12].



**6 pav.** Santykinis trumpalaikių, vidutinės trukmės ir klasikinių prognozių tikslumo palyginimas. Šaltinis: [12]

## 2.5 Mašininis mokymasis ir dirbtiniai neuroniniai tinklai

Mašininis mokymasis – tai kompiuterijos šaka, kai kompiuteris sprendžia kokį nors uždavinį nesinaudodamas jokiais iš anksto žinomais algoritmais, o išmoksta sprendimą iš jam pateikiamų apmokymo duomenų ir laukiamų rezultatų. Mašininis mokymasis nėra vien tik išmani sprendimus pagal duomenis įsimenanti sistema: efektyvus mašininio mokymosi algoritmas stengiasi atrasti bendras taisykles ir tendencijas, pagal kurias gaunami rezultatai, tad geba teisingai klasifikuoti ar prognozuoti sprendimus ir pagal tokius duomenis, kuriais nebuvo apmokytas

paruošimo metu.

Mašininio mokymosi metodai plačiai taikomi sprendžiant įvairias realaus pasaulio užduotis, kai sunku sukurti efektyvius taisyklėmis grįstus algoritmus: tai nuotraukų klasifikavimas pagal jose pavaizduotus objektus, šnekamosios kalbos atpažinimas, vertimas, reklamų ar vaizdo įrašų rekomendacijos pagal vartotojo pomėgius, autonominiai automobiliai, orų prognozės, saugumas ir t. t.

### 2.5.1 Mašininio mokymosi tipai

Prieš pradėdant nagrinėti įvairias mašininio mokymosi architektūras, verta susipažinti su pačiu modelio apmokymo procesu. Skiriami du pagrindiniai mašininio mokymosi modelio apmokymo tipai: prižiūrimasis (angl. *supervised*) ir neprižiūrimasis (angl. *unsupervised*).

Prižiūrimojo mašininio mokymosi metodai automatiškai atranda priklausomybę tarp pateiktų objektą apibūdinančių savybių ir norimo rezultato, paremtų anksčiau surinktų istorinių duomenų aibe [13].

Vienas iš svarbių prižiūrimojo mašininio mokymosi porūšių yra skatinamasis mokymasis (angl. *reinforcement learning*). Šio mokymosi metu modeliui nekeliami jokie konkretūs tikslai: nurodoma tik tai, ar modelis, priimdamas vienokį ar kitokį sprendimą, pasielgė teisingai. Šis metodas gerai tinka dinaminėje aplinkoje, kai pradžioje nėra visiškai aišku, kokie sprendimai ar veiksmai reikalingi, norint pasiekti geriausią rezultatą. Tiesa, tokiu būdu produkcinėje aplinkoje veikiantis ir nuolat apmokomas modelis gali prarasti savo tikslumą, tad ne visada gali būti patikimas.

Neprižiūrimąjį mašininį mokymąsi J. D. Kelleher'is et al. savo knygoje apibūdina kaip mokymąsi, besiremiantį stebėjimais ir atradimais. Modeliui nesuteikiamos jokios tam tikrą duomenų aibę apibūdinančios sąvokos, ir pats modelis vienu metu gali stebėti daugiau nei vieną savybę [13].

Neprižiūrimojo mokymosi metodai gali būti naudojami vidinių objektų savybių išskyrimui (angl. *feature extraction*), vėliau šiuos rezultatus panaudojant prižiūrimojo mašininio mokymosi modelių įvestyse.

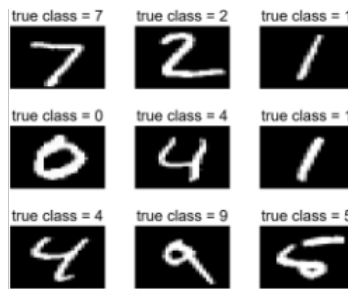
Trumpalaikio kritulių prognozavimo atveju patikimiausias būtų prižiūrimasis mašininio mokymosi metodas, kai apmokytas tinklas produkcinėje aplinkoje išlieka statiškas, t. y. nekeičia savo vidinės struktūros. Tiesa, ateityje būtų galima įvertinti ir skatinamojo apmokymo galimybę, papildant modelį naujai gautais meteorologinio radaro duomenimis.

### 2.5.2 Klasifikacija ir regresija

Mašininiam mokymosi išskiriami du pagrindiniai modelio išvesties tipai: klasifikacija ir regresija. J. Kirchner'is, A. Heberle'as ir W. Löwe'as pateikia tokius šių sąvokų apibrėžimus: klasifikacija susitelkia į priklausomybės tam tikrai klasei nustatymą pagal atributų reikšmes; regresija – į skaitinių reikšmių prognozavimą pagal atributų reikšmes [14].

Paveikslėlyje pavaizduoto objekto priskyrimas tam tikrai klasei yra klasifikacijos uždavinys: pvz., ranka rašyto skaičiaus atpažinimas (žr. 7 pav.). Tuo tarpu, įmonės akcijų vertės smukimo/pakilimo prognozė yra regresijos uždavinys, nes išvestyje reikalinga skaitinė reikšmė.

Mūsų tyrimo atveju reikalinga regresija – kritulių stiprumas kuriame nors taške duotuoju laiko momentu. Tiesa, uždavinys galėtų būti supaprastintas ir iki klasifikacijos, jei būtų norima nustatyti tik faktą, ar norimoje vietovėje lis ar nelis.



7 pav. Ranka rašytų skaičių klasifikacija. Šaltinis: Kaggle

### 2.5.3 Dirbtiniai neuroniniai tinklai

Yra sukurta įvairiausių mašininio mokymosi algoritmų: sprendimų medžiai, Bajeso tinklai, atraminių vektorių mašinos, dirbtiniai neuroniniai tinklai, tiesinė regresija ir kt. Kiekvienas iš šių algoritmų naudojamas spręsti įvairioms skirtingoms problemoms, tačiau pastaruoju metu dėl išaugusio kompiuterių pajėgumo ir išstobulintų vidinių algoritmų populiariausi yra įvairių tipų dirbtiniai neuroniniai tinklai (DNT).

J. Rosa savo straipsnyje teigia, kad dirbtiniai neuroniniai tinklai yra paremti abstrakčia ir supaprastinta biologinio neurono struktūra. Dirbtiniai neuroni yra sujungiami tarpusavyje ir išdėstomi sluoksniais, suformuojant didelius neuroninius tinklus, kur apmokymas ir tarpusavio jungtys apibrėžia matematinę tinklo funkciją [15].

Vienas neuronas veikia kaip matematinė funkcija, apskaičiuojanti į neuroną ateinančių įvesčių ir kintamojo neurono svorio (angl. *weight*) sandaugų sumą. Ši suma dažnai apdorojama aktyvavimo funkcija, apribojančia neurono išvestį tam tikrame intervale ir susumuojama su neurono polinkiu (angl. *bias*), (1):

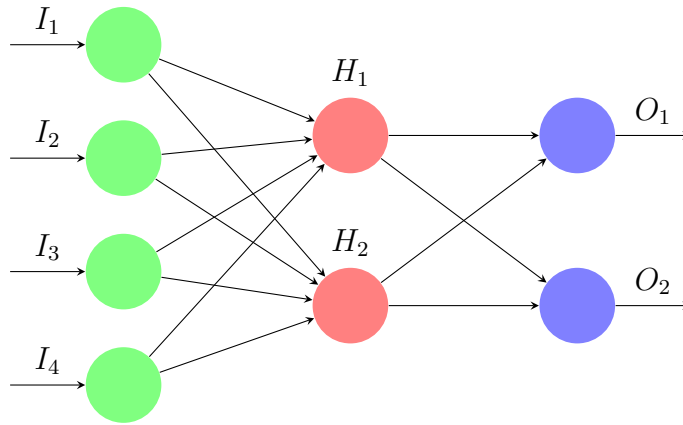
$$f\left(b + \sum_{i=1}^n x_i w_i\right) \quad (1)$$

Čia  $f$  – aktyvavimo funkcija;  $x$  – neurono įvestis;  $w$  – neurono svoriai;  $b$  – neurono polinkis.

Dirbtinis neuroninis tinklas susideda iš mažiausiai trijų sluoksnių su įvairiu kiekiu neuronų: įvesties  $I$ , vieno arba daugiau paslėptųjų sluoksnių  $H$  bei išvesties sluoksnio  $O$ . Pilnai sujungto neuroninio tinklo atveju kiekvieno neurono išvesties reikšmė perduodama visiems sekančiame sluoksnyje esantiems neuronams (žr. 8 pav.).

Neuroninio tinklo apmokymas vyksta mažinant neatitikimą tarp tinklo išvesties reikšmės ir norimo rezultato (angl. *label*) (apibrėžiama klaidos funkcija), atitinkamai pakoreguojant gilesniuose sluoksniuose esančių neuronų kintamųjų – svorių ir polinkių – reikšmes. Šiam veiksmui paprastai naudojamas klaidos skleidimo atgal algoritmas (angl. *backpropagation*), siekiantis gradientinio nusileidimo metodu surasti klaidos funkcijos minimumą.

Bene geriausių rezultatų mašininio mokymosi srityje leido pasiekti gilieji neuroniniai tinklai (angl. *deep neural networks*), t. y. dirbtiniai neuroniniai tinklai, sudaryti iš daugiau nei vieno



8 pav. Pilnai sujungto dirbtinio neuroninio tinklo struktūra

paslėpto (vidinio) sluoksnio. Egzistuoja įvairūs giliųjų neuroninių tinklų architektūrų tipai, iš kurių populiariausi yra konvoliuciniai bei rekurentiniai neuroniniai tinklai.

A. Gibson’as ir J. Patterson’as savo knygoje aprašo konvoliucinių neuroninių tinklų savybę išskirti aukštesnio lygmens analizuojamo objekto parametrus, pasinaudojant konvoliucijomis (sekančiame sluoksnyje esantis neuronas jungiasi tik su dalimi neuronų prieš tai buvusiame sluoksnyje), ir perduodant šiuos duomenis toliau esantiems sluoksniams (žr. 9 pav.). Ši neuroninių tinklų architektūra gerai tinka objektų paveikslėliuose atpažinimui, natūralios kalbos žodžių bei garsų klasifikavimui [16].



9 pav. Konvoliucinis neuroninis tinklas ir paveikslėlio atpažinimas. Šaltinis: [16]

Konvoliucinių neuroninių tinklų architektūra suteikia puikią galimybę analizuoti paveikslėlius: įvestyje galima pateikti skirtingus to paties paveikslėlio (matricos) kanalus, pavyzdžiui, tris kanalus, atitinkančius *RGB* spalvų paletę. Šia savybe galima pasinaudoti ir kitu būdu: skirtingus kanalus panaudoti skirtingiems laiko žingsniams, pavyzdžiui, keliems tos pačios teritorijos kritulių radaro paveikslėliams.

## 2.6 Rekurentiniai neuroniniai tinklai

Iki šiol aptartos dirbtinių neuroninių tinklų architektūros rėmėsi statiniais duomenimis. Tačiau dauguma procesų vyksta realiuoju laiku: slenkančio kritulių debesies judėjimo kryptis priklauso ne tik nuo jo padėties prieš 15 minučių, tačiau ir nuo visos turimos judėjimo istorijos. Šią problemą sprendžia rekurentinis neuroninis tinklas (angl. *recurrent neural networks*). Ši



architektūra pasižymi ciklais jungtyse tarp neuronų, kas leidžia tinklui turėti savotišką atmintį.

Populiariausias rekurentinių neuroninių tinklų tipas – ilgos trumpalaikės atminties tinklai (angl. *long short-term memory, LSTM*). Paslėptuose sluoksniuose dalis įvesties informacijos atkeliauja iš praėjusio laiko momento. Klaidos funkcija minimizuojama pasinaudojant modifikuotą klaidos skleidimo atgal algoritmo versiją: klaidos skleidimu atgal laike (angl. *backpropagation through time*).

Šie tinklai plačiai naudojami įvairiose srityse, modeliuojančiose duomenų pokyčius laike, kai kiekvienas duomenų rinkinys priklauso nuo prieš tai buvusio: tai kalbos modeliavimas, muzikos generavimas, vertimas, vaizdo įrašų analizė ir pan.

Dėl sudėtingos architektūros ir ciklų rekurentinius neuroninius tinklus (taip pat ir LSTM tinklus) sudėtinga apmokyti: juos sunku paralelizuoti ir pritaikyti apmokymui su šiuolaikinėmis vaizdo plokštėmis. Dėl šios priežasties, nors jų architektūra ir sukurta duomenų besikeičiančių laike prognozei, jų gaunami rezultatai gali nusileisti konvoliucinių neuroninių tinklų architektūroms, laike kintančius duomenis saugojančiomis skirtinguose įvesties kanaluose (kai laiko žingsnių kiekis nėra didelis).

## 2.7 Mašininio mokymosi taikymas meteorologijoje orų prognozavime

Mašininio mokymosi pritaikymas prognozuojant orus yra plačiai tiriamas nuo pat tų laikų, kai tik išsivystė ši mokslo šaka. Mašininio mokymosi ekspertai analizavo įvairių šios srities algoritmų ir metodų pritaikymą temperatūros, kritulių, įvairių stichinių reiškinių prognozėms ir pasiekė labai gerų rezultatų, neretai viršijančių tradicinių metodų tikslumą. Šiame skyriuje analizuojami mašininio mokymosi specialistų moksliniai darbai meteorologijos ir orų prognozavimo srityje ir vertinama jų rezultatų įtaka kuriamam tiriamajam darbui.

M. Holmstrom'as, D. Liu ir C. Vo savo straipsnyje „Mašininio mokymosi pritaikymas orų prognozėms“ atlieka tiesinę ir funkcinę regresiją paremtą orų prognozės tyrimą. Prognozei buvo naudoti šie pradiniai duomenys: aukščiausia bei žemiausia temperatūra, vidutinis drėgnis, vidutinis atmosferos slėgis ir oro klasifikacija kiekvienai 2011 – 2015 m. dienai Stanforde, Kalifornijoje [17].

Day	Linear Regression	Functional Regression	Professional
1	5.039	5.252	2.612
2	5.157	5.734	3.244
3	5.300	5.914	3.618
4	5.379	6.068	3.708
5	5.446	6.221	4.522
6	5.566	6.211	4.883
7	5.642	6.329	5.062

**10 pav.** Regresijos metodų ir profesionalių prognozių vidutinės kvadratinės paklaidos palyginimas [17]

Autoriai atliko gautų dviejų regresinių modelių bei profesionalių orų prognozių tikslumo

rezultatų palyginimą (žr. 10 pav.). Kaip palyginimo kriterijus naudojama vidutinė kvadratinė paklaida (angl. *root mean squared error*, *RMSE*). Deja, profesionaliųjų prognozių tikslumas lenkė gautų modelių tikslumą, nors ilgėjant prognozės trukmei klaidų reikšmės beveik susilygina [17].

Pagal šiuos duomenis galima teigti, kad tiesinė regresija nėra pats tinkamiausias mašininio mokymosi metodas orams prognozuoti: ji nesugeba efektyviai įvertinti sudėtingų įvairių oro parametrų sąsajų, tad šiam tyrimui gali būti tinkama tik kaip palyginamasis etalonas.

Kaip jau buvo aptarta anksčiau, orų prognozė gali būti panaudojama įvairių pavojingų stichinių reiškinių spėjimui, tačiau skaičiavimai turi būti atliekami pakankamai greitai: kitaip apskaičiuoti rezultatai gali tapti nebeaktualūs. M. Campolo, P. Andreussi's ir A. Soldati's savo sukurtu metodu straipsnyje „Upių potvynių prognozavimas pasinaudojant neuroninių tinklų modeliu“<sup>5</sup> atkreipia dėmesį į šią problemą [18].

Dirbtinio neuroninio tinklo apmokymui autoriai naudojo pastarųjų 20-ies metų istorinius potvynių duomenis. Be informacijos apie iškritusių kritulių kiekį, buvo panaudoti ir higrometro pagalba surinkti duomenys apie oro drėgnį. Autoriai teigia, jog norint apskaičiuoti upių vandeningumą, vien tik duomenų apie lietų nepakanka: svarbi ir dabartinė oro situacija regione. Pats dirbtinio neuroninio tinklo modelis buvo sudarytas iš neuroninio tinklo, kuriame duomenys perduodami viena kryptimi, be ciklinių jungčių (angl. *feed-forward neural network*) [18].

Straipsnio autoriai buvo itin patenkinti gautais rezultatais: mažiausia 1-os valandos trukmės potvynių prognozės paklaida buvo 4%, ir nors tikslumas mažėjo didinant prognozės trukmę, rezultatų tikslumas išliko patenkinamas iki 5 valandų [18].

T. Hall'as, H. E. Brooks'as ir C. A. Doswell'as, savo ruožtu, atliko tikimybinę ir kiekybinę kritulių prognozę, kurios rezultatus aprašė straipsnyje „Kritulių prognozė pasinaudojant neuroniniu tinklu“<sup>6</sup>. Jie išreiškė abiejų prognozių tipų svarbą: tikimybė yra svarbi tiems, kuriems reikia žinoti, ar kurią nors dieną apskritai lis, o kiekybinė gerai tinka prognozuojant staigius potvynius [19].

Įvesties duomenims buvo naudota 19 skirtingų meteorologinių kintamųjų. Projektuojant neuroninį tinklą buvo svarbus jo pritaikomumas visiems metams, nepriklausomai nuo sezono: skirtingi tinklai kurti šiltajam ir šaltajam metų laikams. Be to, kuriami modeliai buvo interaktyvūs: naudotojas galėjo koreguoti įvesties duomenis ir stebėti, kokią įtaką tai turi prognozės rezultatams [19].

Pasiekti rezultatai buvo kokybiški: tinklui užtikrintai (su aukšta spėjamo rezultato verte išvestyje) prognozuojant, kad kurią nors dieną lis arba nelis, jis dažniausiai būdavo teišus (beveik 100% tikslumu). Kiekybinės prognozės atveju koreliacijos koeficientas tarp realaus iškritusių kritulių kiekio bei prognozuotų reikšmių buvo 0.95. Be to, buvo pastebėta, jog šaltojo sezono prognozės buvo tikslesnės už šiltojo [19].

Šiame modelyje pabrėžiamas interaktyvumas – labai įdomi savybė, kuri realizuojama ir šio tyrimo rezultatų pristatymo sistemoje (žr. 3. Projektinė dalis). Be to, verta įvertinti ir sprendimą atskirti šiltojo ir šaltojo sezonų modelius.

<sup>5</sup>River flood forecasting with a neural network model

<sup>6</sup>Precipitation Forecasting Using a Neural Network

E. Hernandez'as et al. savo straipsnyje „Lietaus prognozė: giliojo mokymosi sprendimas“<sup>7</sup> pristato suminę sekančios dienos kritulių prognozę, pasinaudodami įdomia neuroninio tinklo architektūra, sudaryta iš dviejų dalių: autoenkoderio tinklo bei pilnai sujungto neuroninio tinklo. Autoenkoderis remiasi neprižiūrimojo mašininio mokymosi technologija, kad išskirtų esmines orų elementų savybes, kurios yra toliau perduodamos daugiasluoksniui tinklui [20].

Modeliui apmokyti naudoti net 47 skirtingi orą aprašantys kintamieji (temperatūra, santykinis drėgnis, saulės ryškumas ir kt.) ir 4216 duomenų rinkinių, iš kurių 2952 buvo naudoti apmokymui, o po 632 validacijai ir testavimui.

Gauti rezultatai buvo trečdaliu geresni, nei pasiekti panašiais mašininio mokymosi metodai (vertinant vidutinę kvadratinę paklaidą). Tiesa, dienomis, kai kritulių kiekis buvo nedidelis, tikslumas krisdavo [20]. Toks rezultatas atrodo neintuityvus - jei kritulių nedaug, tikslumas, rodos, turėtų būti geresnis, nes mažesnį kiekį kritulių prognozuoti lengviau. Šią išvadą bus galima patikrinti ir šiame darbe.

Meteorologinio radaro duomenys taip pat plačiai naudojami kritulių prognozėms mašininio mokymosi pagalba. Y. Chiang'as et al. savo straipsnyje „Dinaminis dirbtinis neuroninis tinklas kritulių spėjimui ir prognozei pagal radaro stebėjimus“ aprašo rekurentinių neuroninių tinklų panaudojimą trumpalaikiai lietaus kiekio prognozei taifūnų sezono metu (1-os valandos trukmės prognozėms). Neuroninio tinklo apmokymui naudojami įvairūs neapdoroti meteorologinio radaro surenkami kritulių parametrai [21].

Neuroninis tinklas be ciklinių ryšių šiuo atveju netiko, nes tokio tipo architektūra pasižymi prastais rezultatais, kai dirbama su dinamiška laike kintančia sistema – buvo panaudotas rekurentinis neuroninis tinklas (žr. 2.6 poskyrį), įvertinantis tinklo būseną praeityje. Apmokymui buvo naudotas realaus laiko rekurentinio mokymosi algoritmas, kuris realiu laiku geba atnaujinti savo signalo apdorojimo funkciją pagal naujus duomenis. Neuronų kiekis ir kiti hiperparametrai buvo nustatyti bandymų ir klaidų metodu.

Gauti rezultatai nedaug skyrėsi nuo faktiškai iškritusių kritulių (iki 10 mm) Tiesa, autoriai teigia, kad tikslumas galėjo būti geresnis, nes tinklui apmokyti jie naudojo vos 465 skirtingus duomenų rinkinius. Jie daro išvadą, jog tokiai sudėtingai sistemai toks duomenų kiekis nepakankamas [21].

Taigi, mūsų tyrime svarbu surinkti kuo daugiau kokybiškų duomenų rinkinių. Kaip teigiama panašių tyrimų išvadose, duomenų trūkumas gali būti esminė modelio klaidų priežastis.

Tuo tarpu T. Denoeux'as ir P. Rizand'as nenaudoja neapdorotų meteorologinio radaro duomenų, o kaip ir mūsų tyrime, kritulių prognozes siekia atlikti pagal sugeneruotus meteorologinio radaro duomenų paveikslėlius. Šį metodą jie aprašo straipsnyje „Radaro paveikslėlių analizė lietaus kiekio prognozei, panaudojant neuroninius tinklus“<sup>8</sup>. Modelyje autoriai naudoja spindulio funkcija paremtus neuroninius tinklus [22].

Siekiant pagreitinti tinklo apmokymo procesą, naudotas paveikslėlių rezoliucijos sumažinimas (nuo 256 x 256 iki 64 x 64). Be to, kiekvieno paveikslėlio pikseliai buvo suglotninti

---

<sup>7</sup>Rainfall prediction: A Deep Learning approach

<sup>7</sup>Dynamic ANN for precipitation estimation and forecasting from radar observations

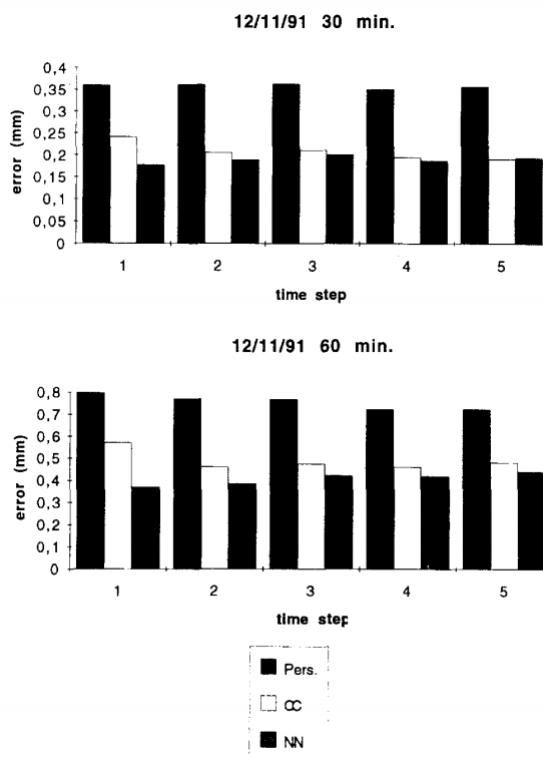
<sup>8</sup>Analysis of Radar Images for Rainfall Forecasting using Neural Networks

(angl. *smoothed*), panaudojant medianos filtrą [22]. Akivaizdu, kad rezoliuciją iki panašaus dydžio mažinti teks ir mūsų tyrime, nes radaro paveikslėlių rezoliucija siekia 796 x 796 pikselius (daugiau apie duomenis - 4 skyriuje). Ir nors dėl tokios drastiškos operacijos prarandama nemažai svarbių detalių, net ir šiais laikais turimų aparatūros pajėgumų neužtektų apmokyti tinklą priimtinu laiku.

Autoriai, vertindami savo algoritmo tikslumą, išskiria skirtingais būdais susidariusius kritulius, matomus radaro paveikslėliuose: susidariusius dėl atmosferos frontų įtakos (radaro paveikslėliuose matomi dideli, stabiliai judantys lietaus plotai) bei dėl konvekcijos (išsibarstę nedideli, gan chaotiškai judantys lietaus debesys).

Straipsnyje taip pat pateikiamas ir įvairių prognozės metodų tikslumo palyginimas (žr. 11 pav.). Stulpelis kairėje atitinka prognozės tikslumą, jei pirminis radaro paveikslėlis nejudėtų (angl. *persistence*); vidurinis – tradicinių ekstrapoliacijos metodų tikslumą (su paveikslėlių tarpusavio koreliacijos koeficientu), dešinysis – neuroninio tinklo gautus rezultatus. Nors tradicinių metodų tikslumas labai panašus į neuroninio tinklo, tačiau pastarasis išsiskiria nežymiai geresniais rezultatais, ypač vertinant mažiausios trukmės prognozes [22].

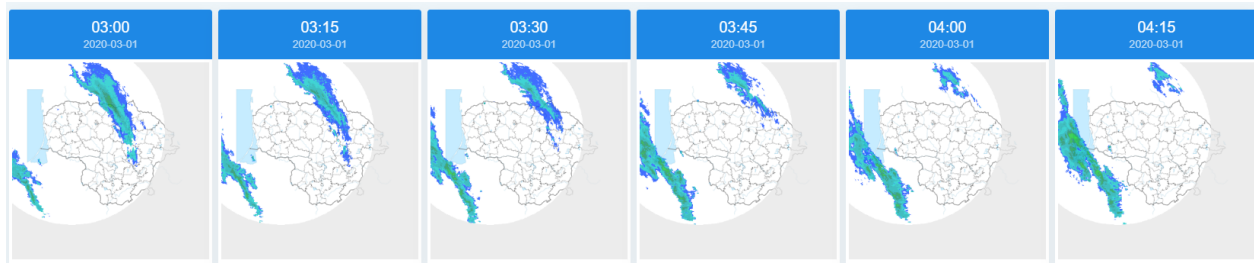
Nors gautų rezultatų lyginimas su sąlygų išsilaikymo matu (angl. *persistence*) atrodo pernelyg primityvus, šis prognozių įvertinimo būdas meteorologijoje yra gana dažnai naudojamas. Meteorologijoje sąlygos nesikeičia taip dažnai kaip gali pasirodyti: jei šiuo metu nelyja (ir apie orus nežinome daugiau jokios informacijos), visai protinga teigti, kad nelis ir toliau; priešingai – jei dabar lyja, didelė tikimybė, kad kurį laiką sąlygos nesikeis ir lyti artimiausiu metu nenustos. Taigi, sąlygų išsilaikymo matas yra tikslesnis nei gali pasirodyti iš pirmo žvilgsnio ir yra geras būdas prognozių tikslumui įvertinti.



11 pav. Prognozės metodų tikslumo palyginimas. Šaltinis: [22]

## 2.8 Mašininio mokymosi taikymas vaizdo įrašų prognozavime

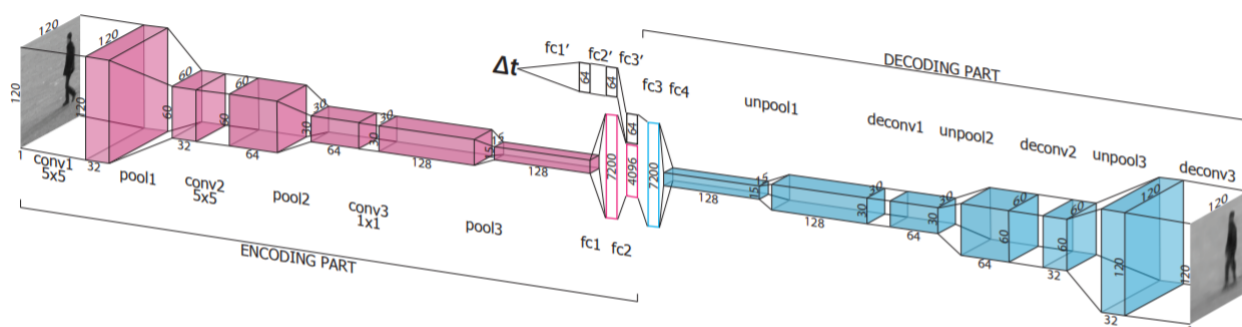
Šio tyrimo užduotis - iš Doplerio kritulių radaro sugeneruotų paveikslėlių prognozuoti kritulius kelioms valandoms į priekį ir rezultatus atvaizduoti kritulių žemėlapiu. Iš esmės, tai yra labai panašu į vaizdo įrašų prognozavimą: iš kelių žinomų kadru bandyti nuspėti kadrus ateityje (žr. 12 pav.).



**12 pav.** Radaro paveikslėlius skirtingu laiku galima įsivaizduoti kaip vaizdo įrašų kadrus, iš kurių galima gauti filmuką

Nors realių fizinio pasaulio objektų ir kritulių judėjimo ypatumai skiriasi, verta susipažinti su naujausiais mašininio mokymosi pasiekimais vaizdo įrašų prognozavimo srityje ir pasisemti žinių apie taikytas architektūras bei metodus, kuriuos būtų galima panaudoti algoritmų konstravimui šiame magistro projekte.

Vukotić'ius et al. savo straipsnyje pristato vaizdo įrašų prognozavimo architektūrą, kuri remiasi konvoliuciniais neuroniniais tinklais [23]. Autoriai nagrinėja žmonių atliekamų judesių prognozavimą vaizdo įrašo kadruose. Jų pristatoma architektūra turi keletą įdomių savybių: įvestyje pateikiamas vienas juodai baltas paveikslėlis, atvaizduojantis žmogaus pozą prieš atliekamą veiksmą; Autoenkoderio struktūros pagalba ši informacija „užkoduojama“ suglaudinta išmokstama reprezentacija. Šioje dalyje kaip papildoma įvestis pateikiamas įvertis  $\Delta t$ , nusakantis, kokios laiko vienetų trukmės prognozė yra atliekama. Galiausiai, ši informacija „at-koduojama“ į pradinės įvesties dydžio paveikslėlį, kuris pateikiamas kaip prognozės rezultatas (žr. 13 pav.).



**13 pav.** Autoenkoderio ir konvoliucinių neuroninių tinklų architektūra žmogaus veiksmo prognozavimui. Šaltinis: [23]

Norint tokią ar panašią architektūrą panaudoti kritulių žemėlapių prognozavimui, reikėtų atlikti keletą pakeitimų. Pristatomame modelyje įvestyje pateikiamas tik vienas kadras, iš kurio

neuroninis tinklas turi išmokyti prognozuoti sekantį atvaizdą ateityje. Tai yra geras sprendimas žmogaus judesių prognozavimui, kadangi iš įvairių žmogaus pozos aspektų modelis gali išmokyti judėjimo kryptį ir kitus prognozei reikalingus parametrus. Su kritulių prognozavimu šiek tiek sunkiau – vargu ar modelis, įvestyje turintis tik vieną paskutinį radaro sugeneruotą kadrą, gebės vien tik iš jo nustatyti kritulių judėjimo kryptį bei stiprumo pokyčio dinamiką.

Lotter'is et al. pristato kitą neuroninių tinklų modelį vaizdo įrašų prognozavimui, kuri įvestyje jau naudoja kelis vieną po kito einančius kadrus [24]. Straipsnyje pristatoma architektūra *PredNet*, kurioje kiekvienas neuroninio tinklo sluoksnis atlieka lokalius spėjimus ir priekyje esantiems sluoksniams perduoda tik nuokrypius nuo šių prognozių. Ši architektūra remiasi hibridiniu CNN ir LSTM (angl. *conv-LSTM*) modeliu.

Straipsnyje pristatomi puikūs rezultatai, pasiekti tiek su sintetiniais vaizdais, tiek su realaus pasaulio pavyzdžiais.

## 3 Projektinė dalis

Internetu galima rasti begalę įvairiausių orų prognozavimo platformų: tiek vietinių, besiorientuojančių į prognozes vienoje šalyje ar regione, tiek pasaulinių.

Absoliuti dauguma orų prognozes pateikiančių tinklapių duomenis gauna iš tų pačių šaltinių: LHMT Lietuvoje, ar meteorologijos giganto *Accuweather* pasaulyje. Dauguma orų prognozių tarnybų naudoja laiko patikrintus analitinius prognozių metodus, kurie remiasi milžinišku pradinųjų duomenų kiekiu ir sudėtingomis diferencialinėmis skysčių bei dujų dinamikos atmosferoje lygtimis.

Šiais laikais kiekvienam, norinčiam sukurti orų prognozes pateikiantį tinklapį, užtenka sumokėti kelis eurus per mėnesį už galimybę naudotis išsamia vieno iš meteorologijos gigantų sukurta ir prižiūrima orų prognozių duomenų baze.

Šiame Magistro projekte kuriamoje programinėje įrangoje nebus naudojamosi jokiomis trečiųjų šalių orų prognozių duomenų bazėmis (išskyrus pradinį Doplerio kritulių radaro duomenimis iš LHMT). Visos prognozės bus generuojamos programinės įrangos viduje, pasinaudojant šio tyrimo metu sukurtais ir išbulintais algoritmais, besiremiančiais mašininio mokymusi.

Tokios kritulių prognozių sistemos, kurioje trumpalaikėms prognozėms naudojami dirbtiniai neuroniniai tinklai, Lietuvoje dar nėra. Be to, kai tyrimo rezultatus galima išbandyti interaktyviai ir praktiškai, galima susidaryti daug geresnį vaizdą apie realų algoritmų tikslumą, nei vien tik skaitant net ir labai detalių gautų rezultatų aprašymą popieriuje.

### 3.1 Sistemos funkcinis aprašymas

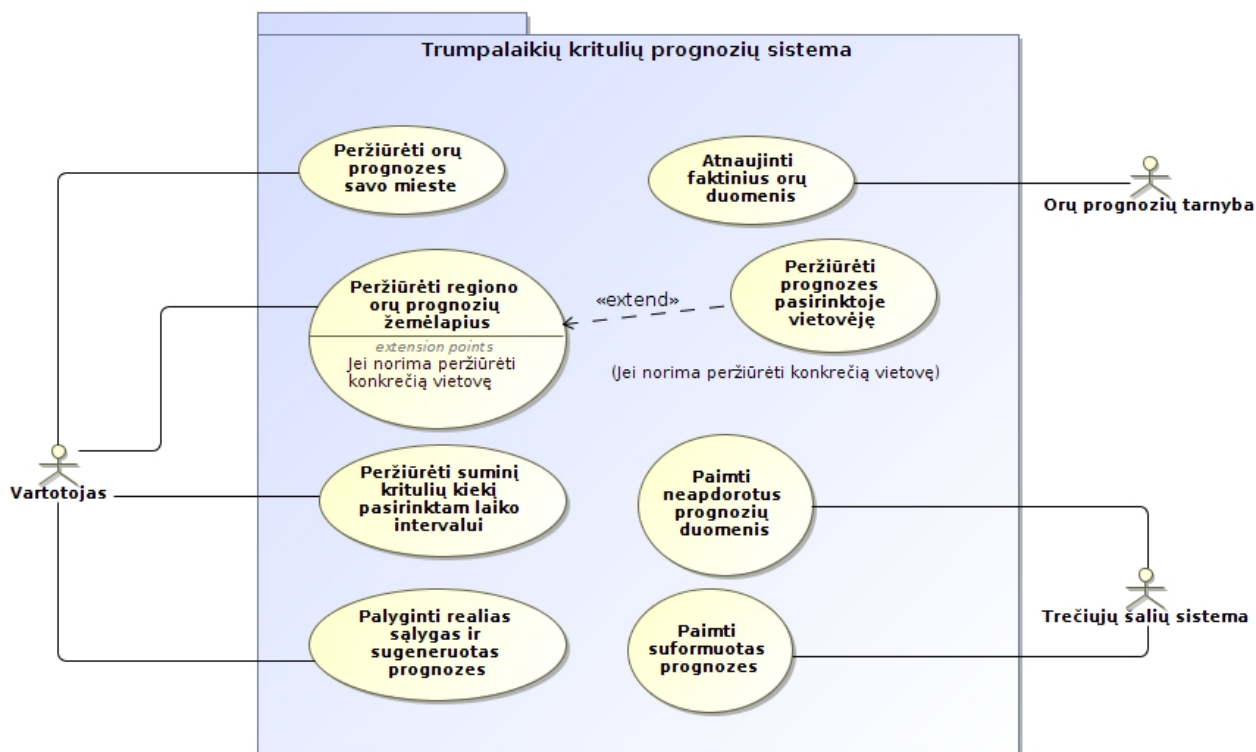
#### 3.1.1 Sistemos funkcijos

Šiame poskyryje aprašomos esminės sistemos funkcijos ir pateikiama formali panaudos atveju diagrama.

Sistema (toliau – Kadalis? sistema), pasiekama internetiniu adresu <https://kadalys.lt>, naudojama trumpalaikėms kritulių prognozėms Lietuvoje peržiūrėti. Pagrindinės sistemos funkcijos ir savybės:

- trumpalaikių kritulių prognozių (iki 4 val.) peržiūra Lietuvos teritorijoje;
- interaktyvūs kritulių žemėlapiai, apimantys visą Lietuvos teritoriją;
- skaitinės kritulių prognozės su informacija apie laiką iki kritulių pradžios ar pabaigos konkrečiame mieste;
- detalios (kas 15 min.) kritulių prognozės daugiau nei 100-ai Lietuvos miestų ir miestelių;
- nuolat tobulinami kritulių prognozavimo algoritmai, paremti mašininio mokymusi;
- istoriniai prognozių ir dienos kritulių stebėjimų duomenys;
- suminių kritulių žemėlapiai pasirinktam laiko intervalui;
- laisvai prieinamas prognozių API.

Sistemos funkcionalumą formaliai galima atvaizduoti panaudos atveju diagrama (žr. 14 pav).



14 pav. Panaudos atvejų modelis

### 3.1.2 Sistemos kontekstas

Šiame poskyryje pateikiamas sistemos kontekstas: galimi PĮ vartotojai, sąveika su trečiųjų šalių sistemomis, kurios gali bendrauti su kuriamą PĮ, ir kurių duomenys potencialiai gali būti naudojami kaip programinės įrangos įvestis.

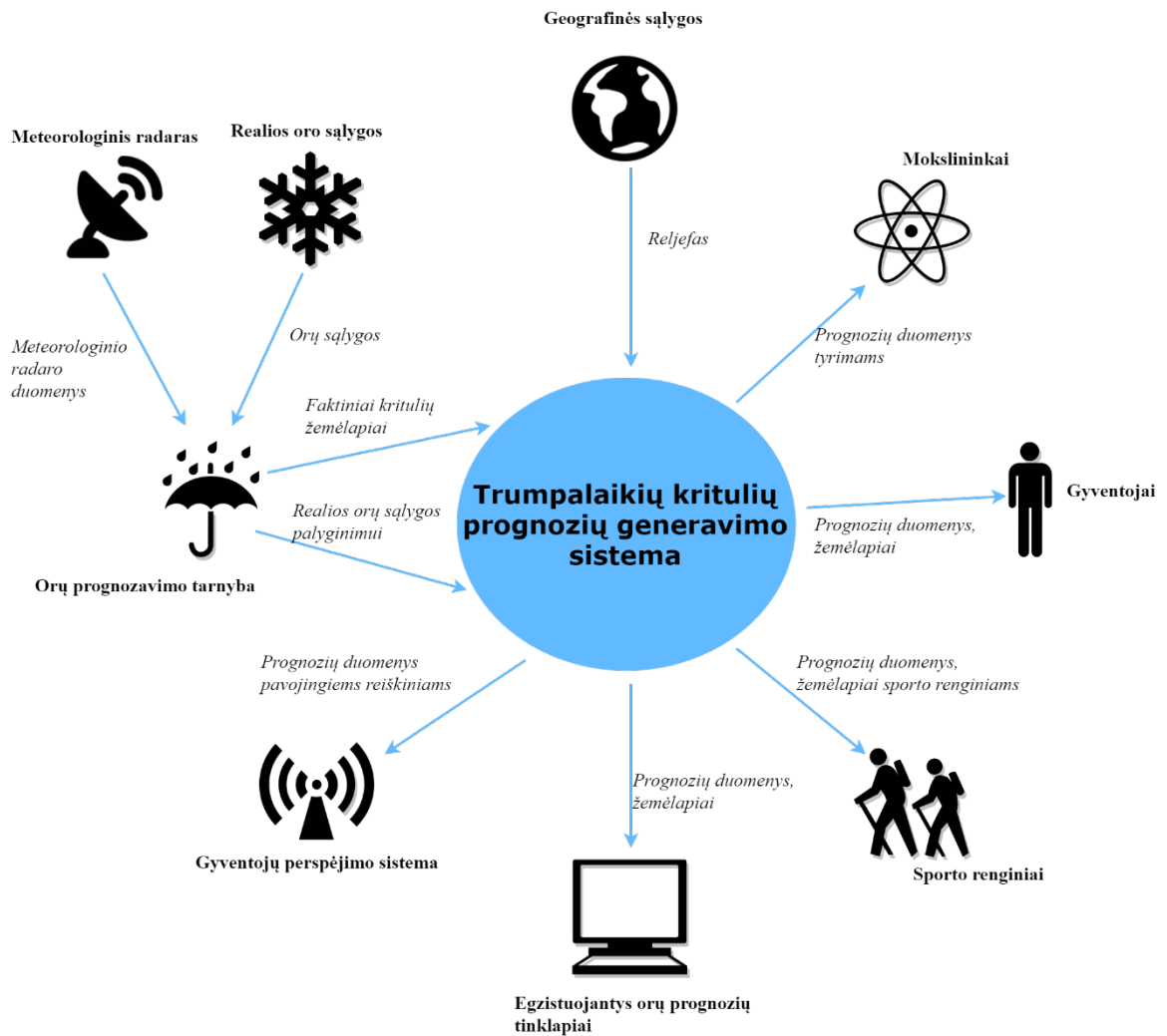
15 pav., 31 puslapyje, pateikiama sistemos konteksto diagrama.

Iš konteksto diagramos galima išskirti šias sistemos vartotojų grupes:

- **gyventojai** – domisi bendromis kritulių prognozėmis savo gyvenamojoje vietoje. Vartotojų grupės kompiuterinis raštingumas gali stipriai varijuoti nuo žemo iki labai aukšto;
- **egzistuojantys orų prognozių tinklapiai** – sukūrus kokybišką sistemą, gali atsirasti norinčių ją integruoti į jau populiarumo sulaukusius tinklapius;
- **gyventojų perspėjimo sistemos** – gyventojų perspėjimo sistemos specialistai sugeneruotas prognozes gali panaudoti galimų pavojingų reiškinių nustatymui;
- **sporto renginiai** – trumpalaikės kritulių prognozės gali būti labai svarbios sporto ar kitų renginių organizatoriams ir dalyviams, nes nuo jų gali priklausyti rungtynių rezultatus ar susirenkančių žmonių kiekis;
- **mokslininkai** – surinkti duomenys gali padėti klimato ar pavojingų reiškinių susidarymo sąlygų tyrimams. Vartotojų kompiuterinis raštingumas labai aukštas.

Priklausomai nuo prognozių pateikimo būdo, norint pasinaudoti prognozių rezultatais nebus reikalingas aukštas kompiuterinis raštingumas, tačiau gali būti pateikiama ir papildoma informacija, skirta vidutiniu ar aukštu kompiuteriniu raštingumu pasižymintiems vartotojams.





15 pav. Sistemos konteksto diagrama

### 3.2 Sistemos apribojimai

Iš 2-ame skyriuje atlikto tyrimo išplaukia keletas sistemos apribojimų, kurie gali turėti įtakos pradiniam PĮ duomenims ar rezultatams.

Visų pirma, dėl naudojamų radaro duomenų ribotumo sistemoje nebus galimybės atskirti kritulių tipo: lietaus, sniego ar krušos. Paprastumo dėlei, visi krituliai sistemoje bus vadinami lietumi, nes tai yra dažniausiai pasitaikantis kritulių tipas. Tiesa, vartotojams turėtų būti pateikiamas šis netikslumą paaškinantis pranešimas (bent jau žiemos mėnesiais).

PĮ veikimas labai priklauso nuo pagrindinio duomenų šaltinio – LHMT sistemos pasiekiamumo ir stabilumo. Patirtis rodo, kad esant didelei apkrovai, LHMT sistema su padidėjusiais vartotojų srautais susitvarko itin sunkiai, tad su panašiomis problemomis tikriausiai susidurs ir kadalis.lt sistema. Deja, neturint jokios tarpusavio bendradarbiavimo sutarties ir naudojantis viešai pateikiamais duomenimis be atlygio, kokių nors ypatingų sąlygų tikėtis neverta ir geriausia susikoncentruoti į tai, ką galima lemti: jei naujausi kritulių radaro duomenys nepasiekiami, turi veikti bent jau prognozių istorijos ir palyginimo funkcionalumas.

Atsižvelgiant į kitus iš kritulių radaro technologijos išplaukiančius apribojimus (žr. 2.3.3 skyrių) - mažą kritulių padengimo rezoliuciją kraštiniuose taškuose ir matomumo lauką užsto-

jančius objektus užtektų apsiriboti informaciniu pranešimu vartotojams, arba apskritai nesiimti jokių veiksmų, nes šių apribojimų įtaka PĮ yra pakankamai maža.

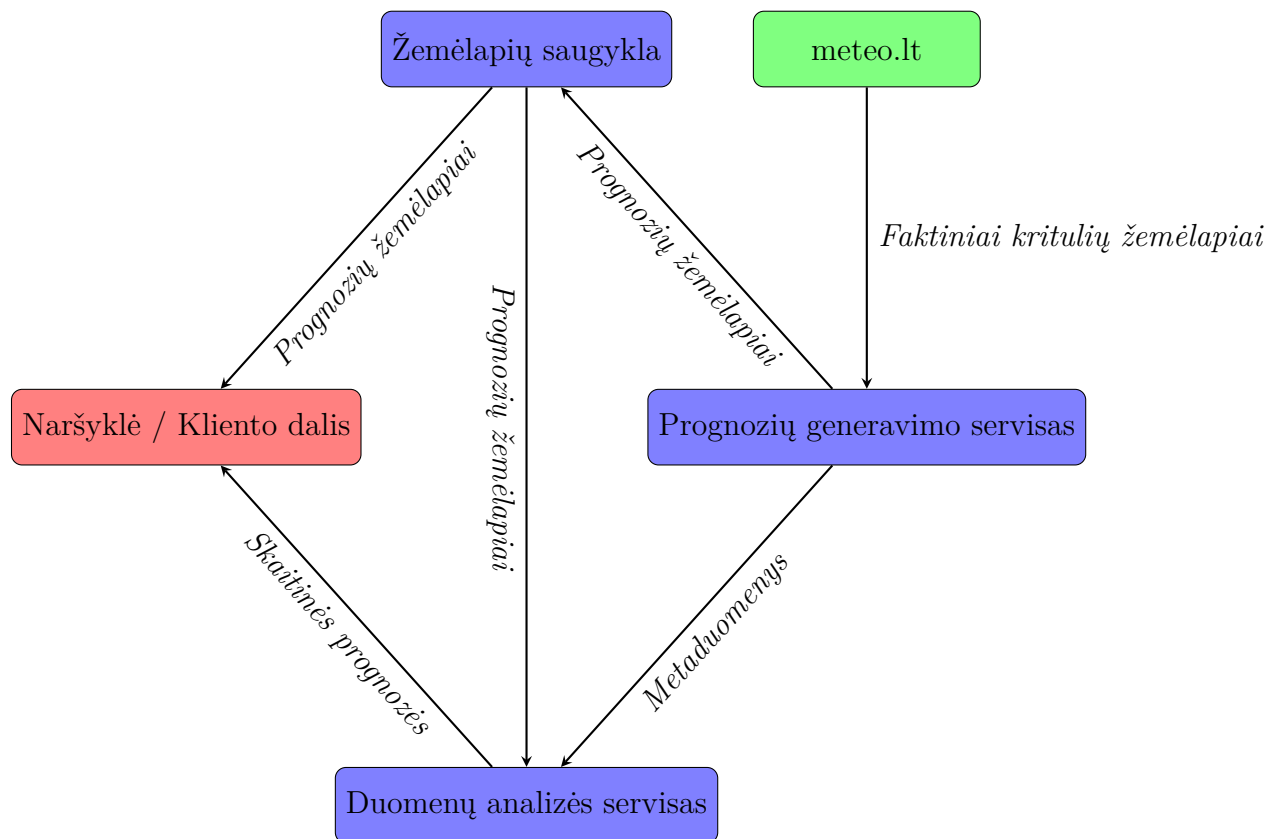
Kitas dalykas į kurį reikia atsižvelgti ir informuoti vartotojus yra ribotas prognozių tikslumas augant prognozės trukmei. Kaip ir analitinėse prognozėse, taikant trumpalaikių prognozių technologijas, jų tikslumas sparčiai mažėja, kai prognozuojama toliau į ateitį. Kai kurios prognozės tampa ne pakankamai tikslios jau po 2 valandų [12]. Visa informacija mūsų svetainėje turėtų būti griežtai ribojama iki 2 – 4 valandų, kad būtų išvengta galimų vartotojų nusiskundimų dėl itin netikslių vidutinės trukmės prognozių.

### 3.3 Statinis sistemos vaizdas

#### 3.3.1 Sistemos sudedamosios dalys

Šiame skyriuje pristatomas statinis kadalis.lt sistemos vaizdas: egzistuojančios posistemės, jų bendravimo kanalai ir pasirinkti architektūriniai sprendimai.

16 pav. pateikiama statinio sistemos vaizdo diagrama.



16 pav. Statinis sistemos vaizdas

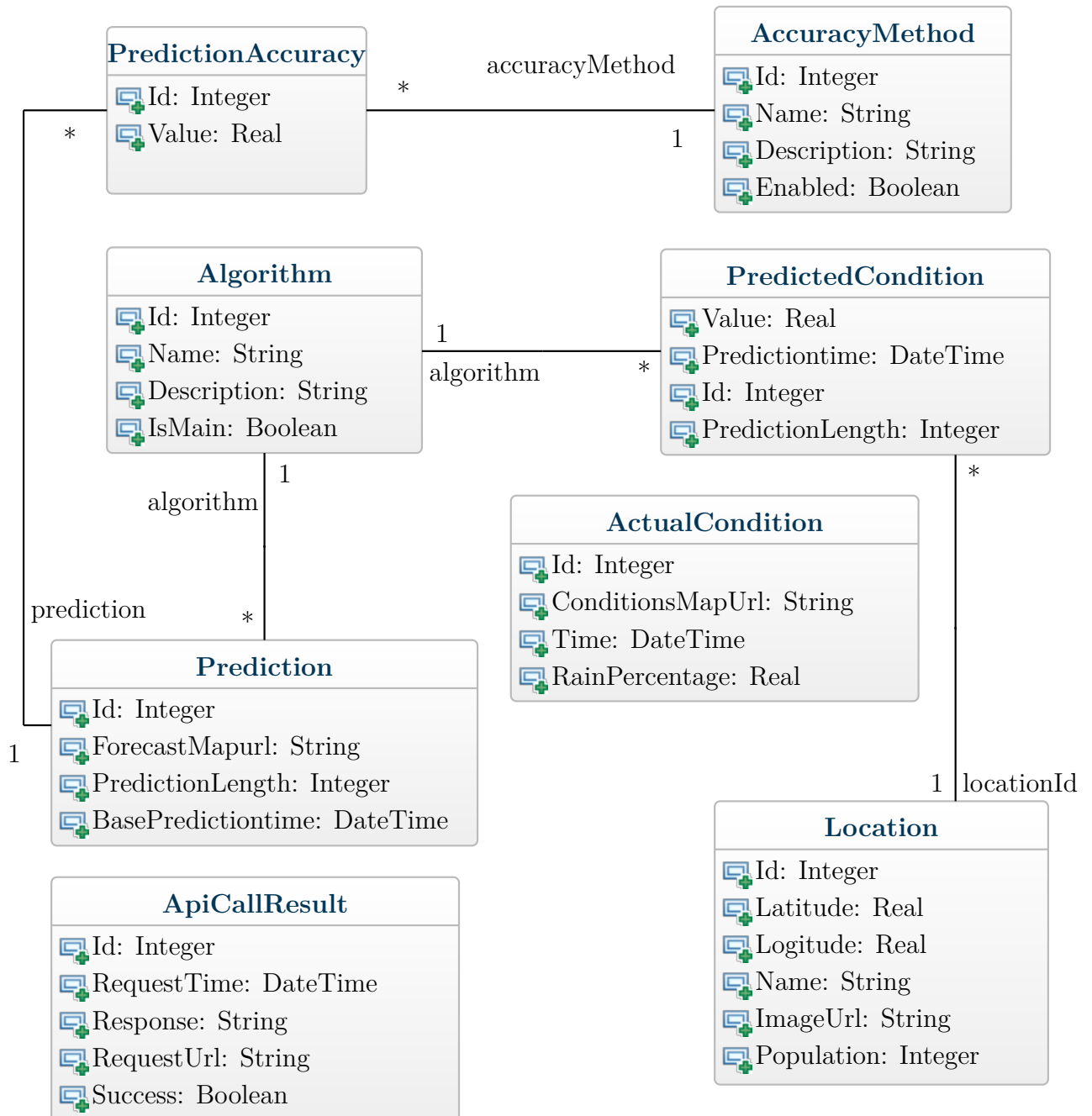
Buvo pasirinktas sprendimas atskirti žemėlapių saugyklą nuo duomenų analizės serverio, nes žemėlapiai sudaro didžiausią sistemos persiunčiamų resursų dalį. Atskyrus saugyklą nuo duomenų analizės serverio, stipriai sumažinama šios svarbios sistemos dalies apkrova, o trečiųjų šalių klientai gali gauti žemėlapius be papildomų metaduomenų, jei šie jiems yra nereikalingi.

Naršyklės/Kliento posistemė atvaizduojama kaip atskira PĮ dalis, nes Kliento dalyje iš tiesių vyksta nemažai papildomų skaičiavimų, tokių kaip prognozių tikslumo įverčio apskaičiavimas

ar suminių kritulių generavimas iš duomenų analizės serviso pateikiamų duomenų.

Diegiant serverio dalį pasirinkta strategija, kai mokama ne fiksuota kaina už serverio veikimą dienai ar mėnesiui, o už faktinius sunaudotus procesoriaus ir atminties resursus. Dėl šios priežasties bet kokių neesminių funkcijų iškėlimas į naršyklę taupo PĮ palaikymo išlaidas. Be to, atliekant prognozės tikslumo matavimo veiksmus serveryje, prognozės apdorojimo laikas išaugdavo kone du kartus.

### 3.3.2 Sistemos duomenų modelis



17 pav. Sistemos esybių ryšių diagrama

17 pav. atvaizduojama sistemos esybių ryšių diagrama, iš kurios gaunamas duomenų bazės modelis. Diagramoje galima pastebėti, kad dvi esybės neturi ryšių su kitomis. Tai:

- *ActualConditions* - skirta saugoti faktinius orų prognozių žemėlapius. Su prognozėmis

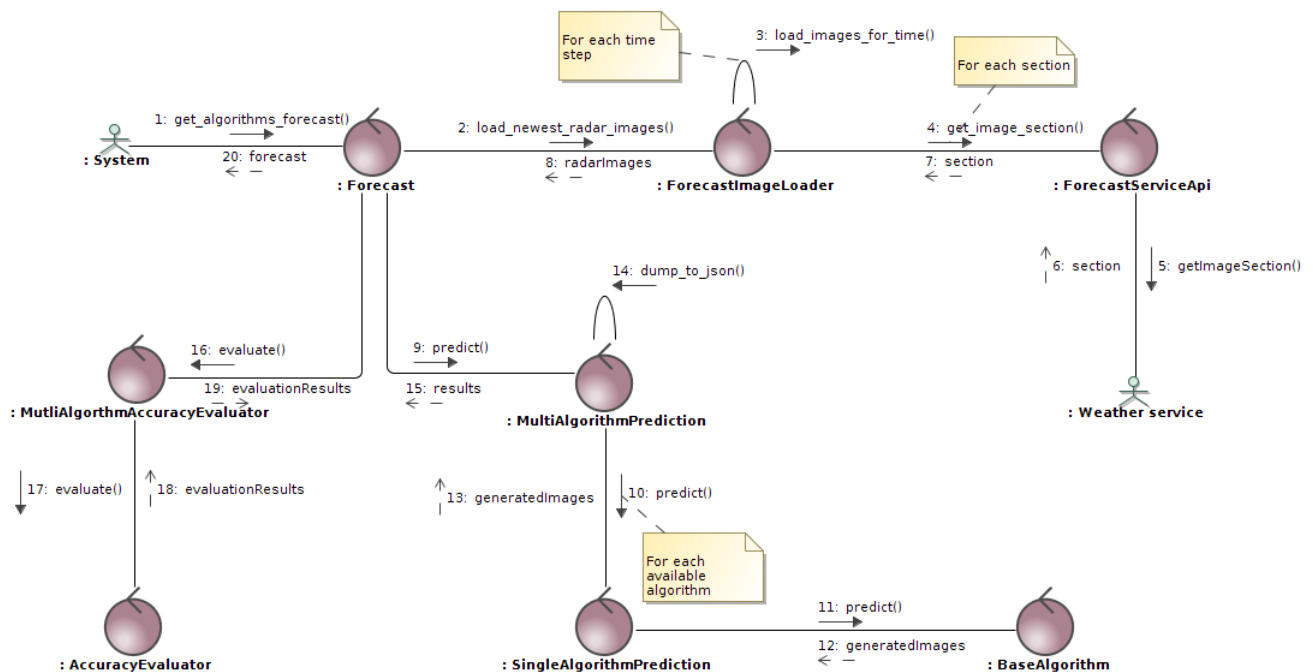
siejama sutampant datai ir laikui, kuriam skirtas žemėlapis;

- *ApiCallResult* - skirta saugoti prognozių generavimo serviso grąžinamus rezultatus bei klaidas vėlesniai jų analizei ir apdorojimui.

### 3.4 Dinaminis sistemos vaizdas

Šiame poskyryje pateikiamas dinaminis sistemos vaizdas, kai vykdomas orų prognozių atnaujinimas iš LHMT. Šiame procese dalyvauja du 3.3 poskyryje apibrėžti servisai: „Prognozių generavimo servisas“ ir „Duomenų analizės servisas“.

Pirmiausia, trečiųjų šalių užklausų planavimo (angl. *scheduling*) servisas kas 15 min. (kas tiek laiko vidutiniškai įvyksta Doplerio kritulių radaro duomenų atnaujinimas LHMT sistemoje) siunčia užklausą į „Prognozių generavimo servisa“. Prasideda veiksmi, pavaizduoti 18 pav., bendradarbiavimo diagramoje.



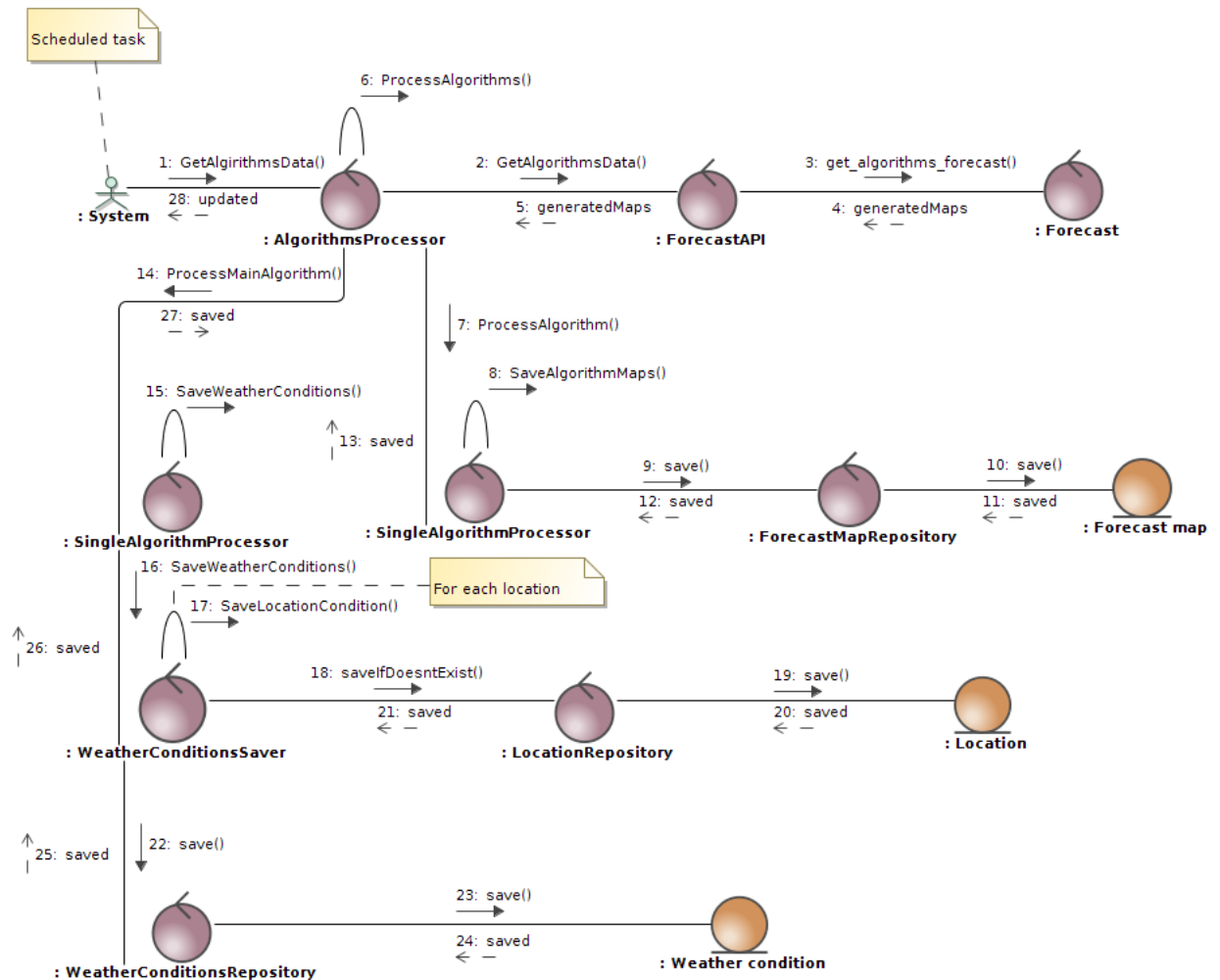
18 pav. Prognozių generavimo serviso bendradarbiavimo diagrama

Iš bendradarbiavimo diagramos galima matyti, kad prognozės generavimo metu atliekami šie veiksmi:

- 1 aktyvuojamas prognozių generavimas;
- 2-3 pradedamas kritulių radaro duomenų atnaujinimas;
- 4-8 atsisiunčiamos ir sujungiamos kiekvieno radaro paveikslėlio dalys;
- 9 pradedamas prognozių generavimas;
- 10-13 kiekvienas aprašytas algoritmas atlieka savo prognozes;
- 14-15 išsaugojami sugeneruotų žemėlapių vardai *.json* formatu;

16-19 įvertinamas prognozių tikslumas (ši dalis prognozavimo metu neatliekama dėl greitaveikos sumetimų, tačiau esant poreikiui gali būti aktyvuojama).

Toliau sugeneruoti rezultatai keliauja į žemėlapių saugyklą, o informacija apie sugeneruotus žemėlapius į duomenų analizės servisą. Čia informacija apie algoritmų sugeneruotus žemėlapius išsaugojama duomenų bazėje (žr. 3.5 skyrių), o patys žemėlapiai naudojami skaitinėms kritulių prognozėms kiekvienai iš apibrėžtų vietovių išgauti (žr. 19 pav).



19 pav. Duomenų analizės serviso bendradarbiavimo diagrama atnaujinant kritulių prognozes

19 pav. pavaizduotoje bendradarbiavimo diagramoje galima išskirti šiuos pagrindinius duomenų apdorojimo veiksmus:

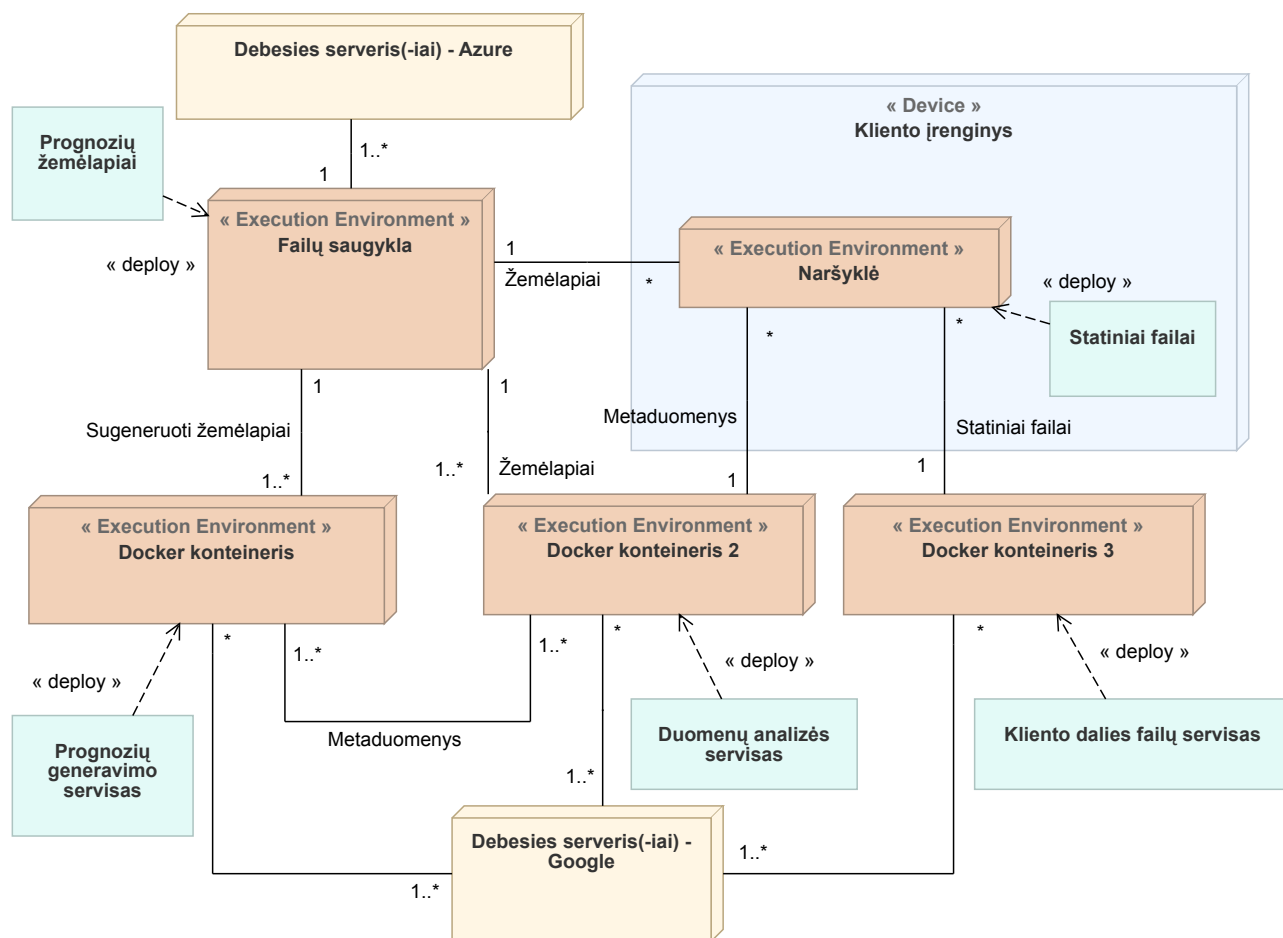
- 1-5 vykdomas prognozių atnaujinimas iš prognozių generavimo serviso (žr. 18 pav.);
- 7-13 duomenų bazėje išsaugojami sugeneruotų prognozių žemėlapių internetiniai adresai;
- 14 apdorojami pagrindinio algoritmo rezultatai;
- 15-16 pradedamas skaitinių orų prognozių išsaugojimas;
- 17-21 pagal žemėlapių pikselių koordinatas, jei reikia, atnaujinamas egzistuojančių vietovių sąrašas;

22-27 iš žemėlapių išskiriamos ir išsaugojamos skaitinės prognozės kiekvienoje palaikomoje vietovėje;

### 3.5 Sistemos diegimas

Sistema įdiegta *Google Cloud* debesyje. Šio debesies diegimo technologija leidžia efektyviai naudoti išskirtus resursus: jei sistema nesinaudojama, visi reikalingi serveriai išjungiami - tokiu būdu taupomos PĮ palaikymo išlaidos.

Pats pagrindinių komponentų diegimas vyksta pasinaudojus *Docker* technologijos palaikoma aplikacijų konteineriais paremta virtualizacija. Kiekvienai vykdymo aplinkai sugeneruojamas virtualus konteineris su visomis reikalingomis operacinės sistemos lygmens bibliotekomis. 20 pav. pateikiama šia idėja besiremianti sistemos diegimo diagrama.



20 pav. Sistemos diegimo diagrama

Diagramoje matyti, kad pagrindiniai servisas yra patalpinti *Google* debesyje, o failų (prognozių žemėlapių) saugykla - *Microsoft Azure* debesyje. *Azure* failų saugykla pasirinkta dėl paprastesnės technologijos, greitaveikos ir subjektyviai patogesnio failų talpinimo. Esant poreikiui, žemėlapių saugykla galėtų būti ir tame pačiame debesyje.

### 3.6 Projektinės dalies rezultatai

Projektinėje dalyje Magistro projekto metu pasiekti šie rezultatai:

- sistema sėkmingai paleista nuo 2019 lapkričio mėn.;

- pasak *Google Analytics* platformos, prie tinklapio per paskutinis 3 mėnesius bent kartą prisijungė 60 unikalių vartotojų;
- 95 % tinklapyje apsilankiusių vartotojų buvo iš Lietuvos, likę – iš Junginės Karalystės, Lenkijos ir kitų valstybių;
- po magistro projekto tyrimo sistema bus papildyta naujais, tikslesniais kritulių prognozavimo algoritmais;
- tikimasi, kad pagerinus prognozavimo algoritmus ir pareklamavus sistemą socialiniuose tinkluose, vartotojų kiekis ir tinklapyje praleidžiamas laikas išaugs.

## 4 Tyrimas

Norint kurti tikslius mašininio mokymosi modelius, kurie mokėtų prognozuoti kritulių judėjimą ir stiprumo dinamiką, būtina suprasti, kokie duomenys yra prieinami: svarbi bendrų charakteristikų analizė, įvairių tendencijų atradimas. Toks turimų duomenų tyrimas gali padėti atrasti įvairius duomenų trūkumus, kurie savo ruožtu gali sąlygoti modelio polinkį (angl. *bias*) prognozuoti klaidingus rezultatus.

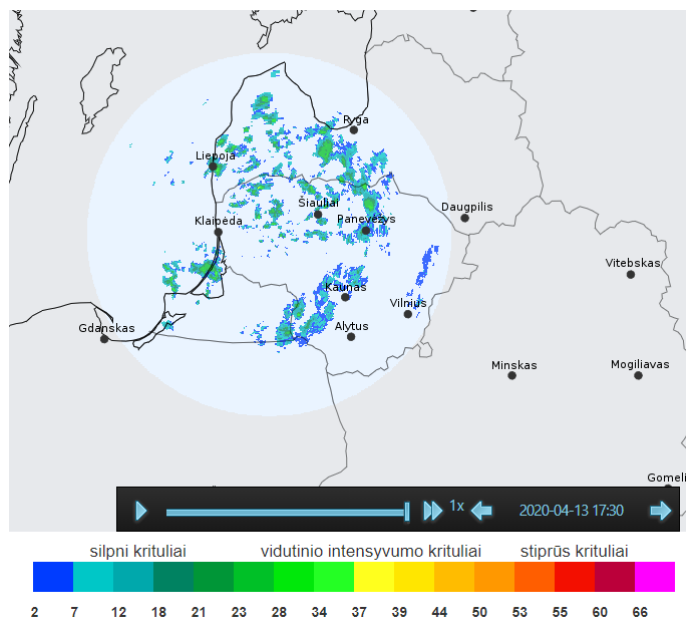
Duomenų analizė skirstoma į šiuos penkis etapus: surinkimas, apdorojimas, tyrinėjimas, modeliavimas ir ataskaitos<sup>9</sup> [25]. Šiame skyriuje bus nagrinėjami pirmi trys žingsniai:

- *surinkimas* (4.1.1 poskyris) – duomenų šaltinis ir bendros prieinamų duomenų charakteristikos;
- *apdorojimas* (4.1.2 poskyris) – duomenų transformacijos ir apdorojimo veiksmai patogiai ir efektyviai duomenų analizei ir paruošimui darbui su mašininio mokymosi modeliais;
- *tyrinėjimas* (4.2 poskyris) – tiriamoji duomenų charakteristikų ir tendencijų analizė.

Šis skyrius apima ir šiame darbe naudojamus prognozių tikslumo matavimo metodus: tradicinius, naudojamus statistinėje analizėje, ir specializuotą, skirtą išimtinai erdviųjų orų prognozių tikslumo įvertinimui (4.3 poskyris).

### 4.1 Duomenys

#### 4.1.1 Duomenų surinkimas



**21 pav.** Viešai prieinama kritulių radaro informacija, 2020-04-13, 17:30

Pradiniai šio tyrimo duomenys yra faktiniai kritulių radaro paveikslėliai, sugeneruoti iš LHMT priklausančių Doplerio kritulių radarų (žr. 2.3.2 poskyrį).

<sup>9</sup>angl. *acquisition, wangling, exploration, modeling, reporting*



Šie duomenys yra prieinami viešai<sup>10</sup> ir atnaujinami apytikriai kas 15 min. (žr. 21 pav.). Skirtingos spalvos kritulių žemėlapyje atitinka kritulių stiprumą: kuo spalva šviesos spektro skalėje arčiau raudonos, tuo radaro užfiksuoti krituliai stipresni.

Tinklapyje pateikiama ir kritulių intensyvumo skalės legenda su skaitinėmis vertėmis, tačiau neakcentuojama kokiais matavimo vienetas pateikiama ši informacija. Žinant dažniausiai pasitaikančias kritulių stiprio matavimo metrikas, galima daryti prielaidą, kad tai yra milimetrai per valandą (*mm/h*), tačiau patvirtinti šią hipotezę be informacijos iš pačios LHMT būtų sudėtinga.

Dėl šios priežasties, tyrime kritulių stiprumas matuojamas kategorijomis, atitinkančiomis skirtingas spalvas kritulių intensyvumo skalėje (žr. 21 pav.). Kategorijos žymimos sveikaisiais skaičiais nuo 0 iki 16, kur 1 atitinka labai silpnus kritulius, 16 – itin stiprius. Skaičiumi 0 žymimos sąlygos be užfiksuotų kritulių.

Tyrime naudojama kvadratinės formos duomenų sritis (parinkta pagal abiejų Lietuvoje veikiančių kritulių radarų dengiamą teritoriją) yra apribota šiais taškais (žr. 22 pav.):

- viršutinio kairiojo kampo geografinės koordinatės – 57° 23' 12.48" šiaurės platumos ir 19° 46' 24.96" rytų ilgumos;
- apatinio dešiniojo – 52° 33' 44.64" šiaurės platumos ir 28° 6' 34.92" rytų ilgumos.

Viena kvadrato kraštinė atitinka apytikriai 500 kilometrų.



22 pav. Geografinis duomenų srities padengimas<sup>11</sup>

LHMT tinklapyje prieinami tik patys naujausi (paskutinių 8–12 valandų) radaro duomenys. Dėl šios priežasties viso tyrimo metu duomenys buvo renkami automatizuotos sistemos, nepažeidžiant jokių autorinių teisių aprašytų tinklapyje<sup>12</sup>. Užklausų kiekis buvo ribojamas remiantis protingo naudojimosi viešai prieinamomis paslaugomis principais.

Apmokymui skirti duomenys apima laikotarpį nuo 2017-10-22 13:45 iki 2019-08-25 14:45. Iš viso **61 373** paveikslėliai. Paskutiniai 1 760 paveikslėlių iš šių duomenų atidedami tarpiniam

<sup>10</sup><http://www.meteo.lt/lt/radaru-informacija>

<sup>11</sup>Šaltinis: <https://earthexplorer.usgs.gov/>

<sup>12</sup><http://www.meteo.lt/lt/autorines-teises>

validavimui, kuris atliekamas kiekvienos modelio apmokymo epochos pabaigoje. Jie yra skirti hiperparametrų optimizavimui. Šie validavimo duomenys naudojami ir pradiniais eksperimentų rezultatams tirti (žr. 5.4 ir 5.5 skyrius).

Testavimui skirti duomenys yra beveik nepertraukiama paveikslėlių seka nuo 2020-01-03 19:15 iki 2020-05-13 17:15. Iš viso – **11 449** radaro žemėlapiai. Atkreiptinas dėmesys į tai, kad faktiškai testavimui naudojamų duomenų kiekis yra mažesnis dėl pasirinkto lietaus padengimo procentinio slenksčio (žr. 5.1.3 skyrių).

Nei apmokymo, nei validavimo, nei testavimo duomenų aibės nepersidengia.

Šie duomenys naudojami tik pačioje eksperimento pabaigoje, vertinant galutinį apmokytų algoritmų tikslumą. Testavimo ir validavimo duomenų atskyrimas yra svarbi bet kokio kokybiško tyrimo metodika, siekianti maksimaliai skaidraus ir tikslaus algoritmų kokybės įvertinimo.

Dėl šios priežasties šiame skyriuje atliekama duomenų analizė remiasi tik apmokymo ir validavimo duomenimis.

#### 4.1.2 Duomenų apdorojimas

Radaro paveikslėliai prieinami  $796 \times 796$  pikselių dydžio raiška. Tokio dydžio duomenų apdorojimui turimi aparatinės įrangos resursai nėra pakankami. Dėl šios priežasties pradiniai duomenys yra sumažinami iki modelių apmokymą smarkiai pagreitinančios  $64 \times 64$  pikselių raiškos. Žinoma, dėl tokios drastiškos operacijos kiekviename paveikslėlyje prarandama nemažai informacijos, tačiau be papildomų resursų didesnių modelių apmokymas užtruktų žymiai daugiau laiko; dėl padidėjusio prognozės taškų kiekio sumažėtų ir prognozių tikslumas. Tiesa, meteorologijos mokslų taikymo srityje tokios duomenų transformacijos yra toleruotinos, kitaip nei, pavyzdžiui, medicinoje, kur bet koks duomenų raiškos praradimas gali būti kritinis [26].

Net ir su sumažinta rezoliucija vienas kvadratinis pikselis radaro paveikslėlyje atitinka apytikriai  $64 \text{ km}^2$  realybėje, kas leidžia Lietuvos teritorijoje sutalpinti daugiau nei 1 000 prognozės taškų. Paveikslėlių dydis pakeičiamas naudojantis artimiausių kaimynų (angl. *Nearest-neighbor*) algoritmu.

Kadangi paveikslėliai sudaryti iš skirtingų spalvų pikselių, atitinkančių įvairaus stiprumo kritulių kategorijas, kiekvienas taškas visuose radaro paveikslėliuose pagal jo spalvą yra konvertuojamas į skaitinę kategorijos reikšmę nuo 0 iki 16.

Aukščiau aprašytais būdais apdoroti kritulių žemėlapių paveikslėliai atvaizduoti 23 pav.

Šio tyrimo metu prognozuojama ne kokios nors statiškos paveikslėlio charakteristikos, o dinamiškas kritulių judėjimas. Dėl šios priežasties duomenis patogiu iš anksto sugrupuoti į egzempliorius, susidedančius iš 4 šaltinio paveikslėlių  $X$  ir 1 rezultato paveikslėlio  $Y$ .

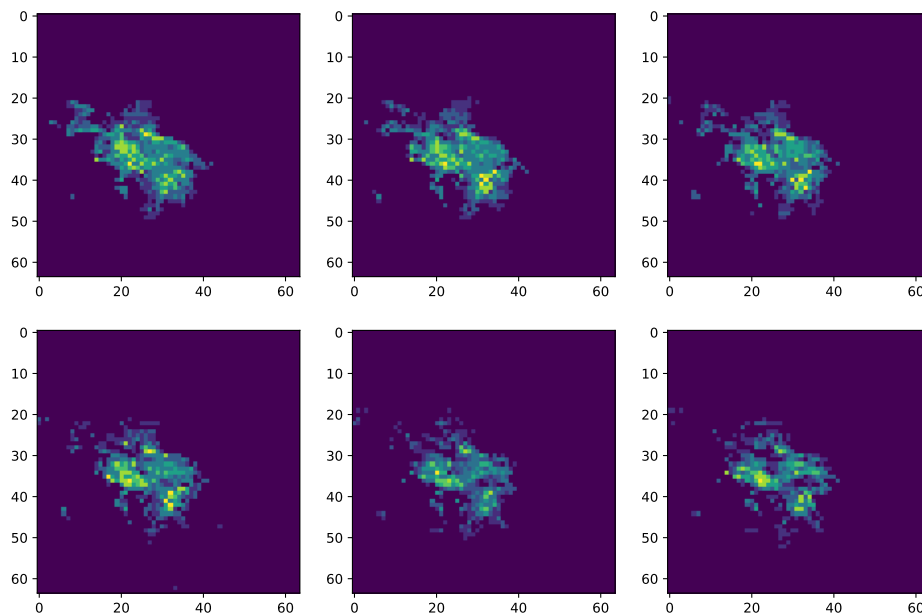
Šaltinio paveikslėliai

$$X = [x_{n-3}, x_{n-2}, x_{n-1}, x_n]$$

yra iš eilės einančių kritulių žemėlapių seka, pateikiama kaip pradiniai modelio duomenys baziniu prognozės momentu  $n$ .

Rezultato paveikslėlis

$$Y = x_{n+1}$$



**23 pav.** Apdorotų radaro paveikslėlių seka, paruošta analizei ir modelių apmokymui

yra kritulių žemėlapis vienu laiko žingsniu po bazinio prognozės momento, kurį modelis turi išmokti prognozuoti.

Dėl įvairių radarų ir duomenų šaltinio veikimo sutrikimų, duomenų surinkimo metu gauti kritulių žemėlapiai nėra vientisa nenutrūkstama seka: duomenyse yra tuščių periodų, kurie neleidžia suformuoti nenutrūkstamos  $X$  ir  $Y$  duomenų eilės. Dėl šios priežasties, modelių apmokymui ir validavimui naudojami 32 432 skirtingi egzemplioriai.

#### 4.1.3 Duomenų saugojimas su retosiomis matricomis

Net ir smarkiai sumažinus paveikslėlių rezoliuciją, jų kiekis vis dar sukelia keletą problemų. Šešiasdešimties tūkstančių atskirų rinkmenų užkrovimas kiekvienos OS failų sistemai yra nemenkas iššūkis, tad duomenų analizei ir lengvam paveikslėlių perkėlimui bei dalinimuisi patogų viską turėti viename faile.

Paveikslėlius galima saugoti atmintį taupančiu *.png* formatu, tačiau jei norime viename faile patogiai patalpinti mokymui paruoštas 4-ių paveikslėlių šaltinio ir 1 paveikslėlio rezultato egzempliorių grupes, tai nėra tinkamas būdas.

Galima tiesiog saugoti paveikslėlių pikselių reikšmių matricas, sujungtas į vieną rinkmeną, tačiau toks failas neglaudintas užimtų apie 5 GB duomenų. Toks kiekis tikrai nėra patogus tyrimo metu nuolat perkeliant ir dalijantis duomenimis.

Atidžiau pažvelgus į duomenis (žr. 4.2 skyrių) galima pastebėti, kad daugumoje paveikslėlių krituliai užima tik nedidelę dalį atvaizdo matricos. Didžioji saugomos informacijos dalis yra tuščių pikselių reikšmės.

Dėl šios priežasties duomenų saugojimui gali būti naudojama retoji matrica. Ši matrica saugoja ne visą elementų sąrašą, o tik nenulinius elementus ir jų koordinatas matricoje. Tokiu būdu taupoma kompiuterio atmintis, kai didžioji dalis duomenų egzempliorių yra tušti.

Naudojant šį duomenų glaudinimo būdą, visi apmokymo ir validavimo duomenys užima

vos 40 MB ir gali būti patogiai perkeltami ir naudojami tiek vietinėje failų sistemoje, tiek ir interaktyviose internetinėse platformose.

## 4.2 Duomenų tyrinėjimas

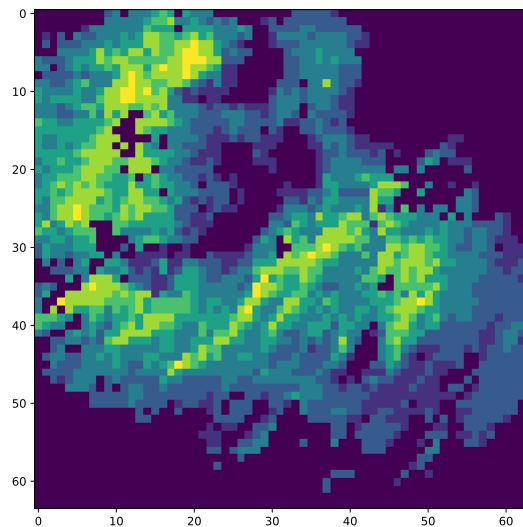
Šiame poskyryje analizuojamos įvairios apmokymo ir validavimo duomenų charakteristikos. Duomenų apimtis: **61 373** paveikslėliai nuo 2017-10-22 13:45 iki 2019-08-25 14:45.

### 4.2.1 Kritulių kiekis ir padengimas

Šiame poskyryje kiekvienam paveikslėliui iš visos duomenų imties matuojamos trys charakteristikos:

- *kritulių padengimas* – kiek procentų visos matomos teritorijos konkrečiu laiko momentu dengė bet kokio stiprumo krituliai (*lietaus stiprumo kategorija > 0*);
- *bendras kritulių stiprumas* – bendra kritulių kategorijų stiprumo kiekviename paveikslėlio taške suma;
- *vidutinis kritulių stiprumas* – vidutinė kritulių stiprumo kategorija, įskaičiuojant tik taškus su bent minimaliu kritulių kiekiu (*lietaus stiprumo kategorija > 0*);

Didžiausias vieno 15 min. periodo kritulių padengimas užfiksuotas 2018 m. rugsėjo 26 d. 14:00 buvo **61,35 %** (žr. 24 pav.). Šis laikotarpis pasižymėjo ir didžiausiu bendru kritulių stiprumu: visų taškų kategorijų reikšmių suma siekė **8 027**.



**24 pav.** Laikotarpis su didžiausiu užfiksuotu kritulių padengimu ir bendru kritulių stiprumu

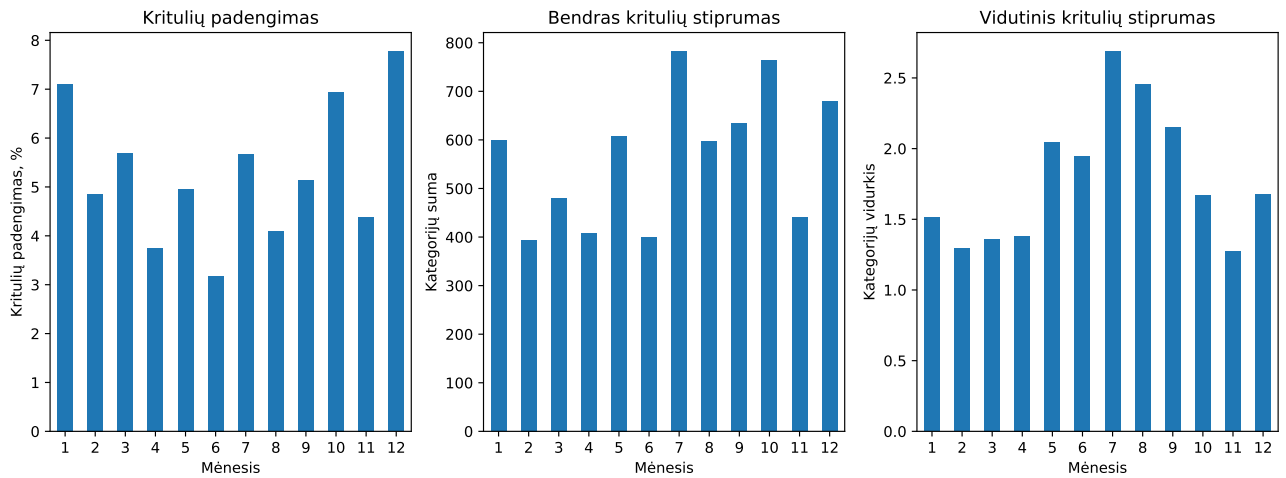
Iš vidutinių kritulių charakteristikų kiekvienam mėnesiui (žr. 25 pav.) galima pastebėti įdomių tendencijų:

- žiemą vyrauja didesnis kritulių padengimo procentas nei vasarą;
- vasarą vidutinis kritulių stiprumas žymiai stipresnis nei kitais metų mėnesiais.

Šios tendencijos gali išryškėti dėl vasarą dažniau pasitaikančių konvekcinių kritulių: tokio tipo kritulių debesys dažniausiai mažesni, apimantys mažesnę šalies teritorijos procentą, tačiau

pašižymi stipresniais krituliais. Šaltuoju metų periodu vyrauja didesnę šalies dalį apimantys silpnesni krituliai.

Šiuos parametrus reikės įvertinti analizuojant eksperimento rezultatus: pvz., ar prognozavimo tikslumas priklausys nuo mėnesio ir vyraujančio kritulių padengimo dydžio.



25 pav. Vidutinės mėnesio kritulių charakteristikos

Kritulių kiekiai buvo išanalizuoti ir pagal paros laiką, tačiau reikšmingų skirtumų tarp kritulių stiprumo dienos ar nakties metu neužfiksuota.

#### 4.2.2 Kritulių judėjimo krypties tendencijos

Vienas esminių mašininio mokymosi algoritmų sėkmės aspektų bus gebėjimas nustatyti kritulių judėjimo kryptį. Dėl šios priežasties verta išanalizuoti bendras kritulių judėjimo tendencijas apmokymui skirtuose radaro paveikslėliuose.

Sudarant trumpalaikes orų prognozes, vienas iš metodų kritulių judėjimo krypties nustatymui yra masės centrų skirtumų analizė [11].

Kritulių judėjimo kryptis apskaičiuojama tokiu metodu: dviejuose kritulių žemėlapiuose, kuriuos skiria 75 minučių laiko tarpas, surandami masių centrai  $C_1$  ir  $C_2$ . Vidutinis visoje teritorijoje matomų kritulių judėjimo vektorius gaunamas radus šių taškų skirtumą:  $\vec{D} = C_2 - C_1$ . Masės centras paveikslėlyje apskaičiuojamas naudojant formulę (2):

$$C_x = \frac{\sum_{i=1}^N m_i x_i}{\sum_{i=1}^N m_i}; \quad C_y = \frac{\sum_{i=1}^N m_i y_i}{\sum_{i=1}^N m_i} \quad (2)$$

čia:  $C_x$  ir  $C_y$  apskaičiuoto masės centro  $x$  ir  $y$  koordinatės;  $N$  – paveikslėlio taškų kiekis;  $m_i$  – kritulių stiprumo kategorijos reikšmė  $i$ -ajame taške;  $x_i$  ir  $y_i$  paveikslėlio taško (pikslio) koordinatės analizuojamame paveikslėlyje.

Prieš atvaizduojant rezultatus gautas kritulių judėjimo vektorius paverčiamas į geografinę pasaulio šalies kryptį, kuri nusako iš kurios pusės ateina krituliai.

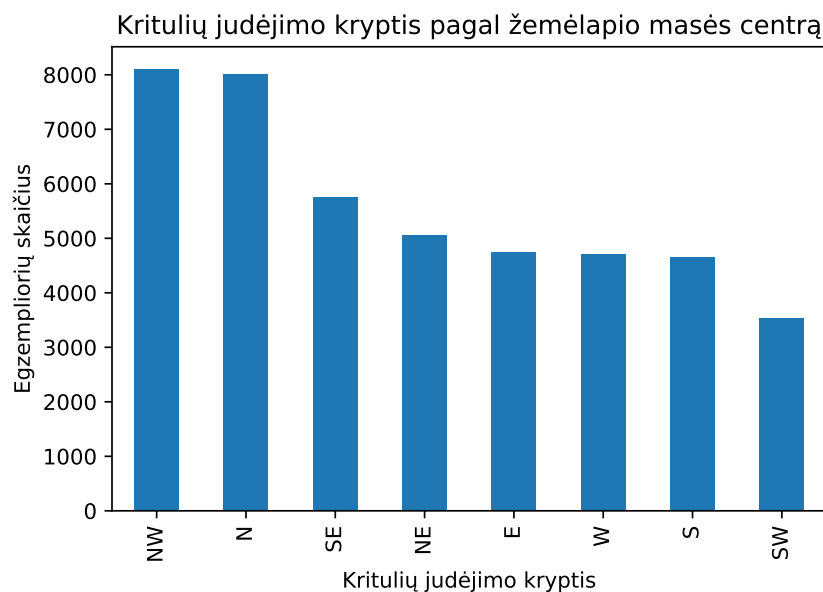
Atlikus visų apmokymui skirtų duomenų analizę, gauti tokie rezultatai: krituliai dažniausiai atkeliavo iš šiaurės vakarų ir šiaurės pusės, dažnai pasitaikydavo ir pietryčių ar šiaurės rytų kryptis. Rečiausiai krituliai judėdavo iš pietvakarių. Pagal dvejų metų Doplerio kritulių radaro duomenis atlikta kritulių judėjimo krypties analizė ne visada sutampa su vidutine daugiamete vėjo judėjimo kryptimi, aptarta 2.2.1 skyriuje: „Lietuvos teritorijoje vyraavo vakarinių ir pietinių rumbų vėjai“ [4].

Vidutinė kritulių judėjimo krypties kiekvienam mėnesiui analizė parodė, kad nuo metų sezono kritulių judėjimo kryptis iš esmės nepriklauso (žr. 1 lentelę). Išskirtinis buvo tik gegužės mėnuo, kurio metu krituliai dažniausiai judėdavo iš pietryčių.

Svarbu atkreipti dėmesį į tai, kad ši kritulių krypties analizė yra tik iliustracinė ir turi trūkumų: analizuota visos virš Lietuvos teritorijos esančios kritulių masės judėjimo kryptis, o ne atskirų kritulių sankaupų dinamika. Be to, analizės kokybę galėjo lemti ir 2.3.3 skyriuje aprašyti radaro technologijos trūkumai.

**1 lentelė** Vidutinė kritulių judėjimo kryptis kiekvieną mėnesį. *N* – šiaurė; *NW* – šiaurės vakarai

Mėnuo	1	2	3	4	5	6	7	8	9	10	11	12
Dažniausia kryptis	NW	NW	N	N	SE	NW	NW	N	NW	NW	N	N



**26 pav.** Dažniausia kritulių judėjimo kryptis, nustatyta pagal kritulių masės centro pokytį

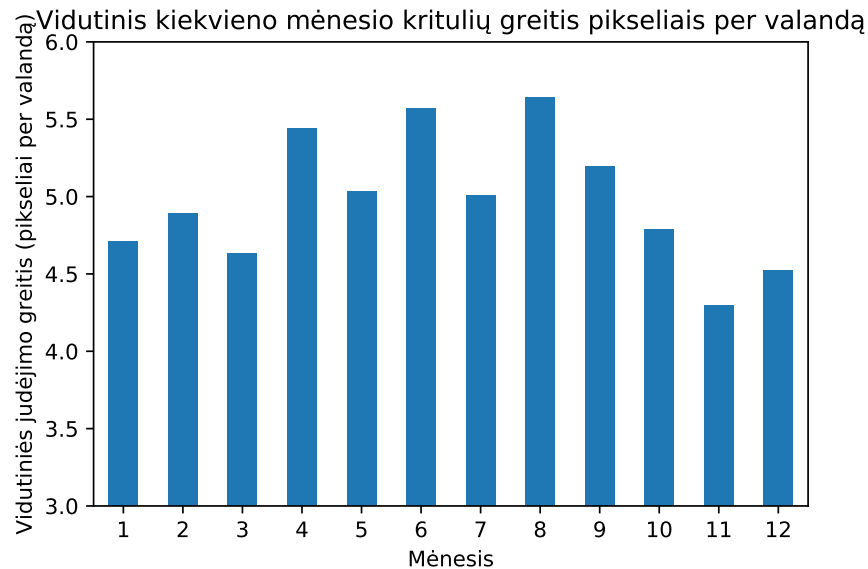
Į prognozuojamą judėjimo kryptį eksperimento rezultatuose verta atkreipti didesnę dėmesį. Dažniausiai pasitaikanti kritulių judėjimo kryptis apmokymo duomenyse gali sąlygoti modelių šališkumą dažniau (ar net visada) prognozuoti kritulių judėjimą šiomis kryptimis.

#### 4.2.3 Kritulių judėjimo greitis

Naudojant 4.2.2 skyriuje aprašyto kritulių judėjimo krypties vektoriaus  $\vec{D}$  ilgį  $|\vec{D}|$ , galima apskaičiuoti vidutinį kritulių judėjimo greitį pikseliais per valandą.

Vidutinis kritulių judėjimo greitis visai duomenų imčiai yra **4,97** pikseliai per valandą (arba  $\sim 38,8 \text{ km/h}$ ); mediana – **3,49** pikseliai per valandą ( $\sim 27,2 \text{ km/h}$ ).

27 pav. pavaizduotas vidutinis kiekvieno mėnesio kritulių greitis pikseliais per valandą. Siekiant išryškinti greičio skirtumus,  $y$  ašies reikšmių sritis prasideda nuo 3 pikselių per valandą.



**27 pav.** Vidutinis kiekvieno mėnesio kritulių judėjimo greitis pikseliais per valandą

Iš 27 paveikslėlio galima matyti, kad krituliai šiltaisiais metų mėnesiais juda greičiau, o šaltaisiais – šiek tiek lėčiau. Skirtumas nežymus.

Kritulių judėjimo greičio žinojimas gali padėti nustatant optimalų konvoliucinio neurono tinklo branduolio dydį (angl. *kernel size*).

### 4.3 Prognozių tikslumo matavimo metodai

Įvairiuose orų prognozių sudarymo servisuose ar interneto svetainėse, viešinančiose tas prognozes, didžiausias dėmesys skiriamas patiems prognozių rezultatams. Sunku lengvai rasti orų prognozių šaltinį, kuris pateiktų bet kokio pobūdžio prognozių tikslumo analizę, t. y. įvertinimą, kiek tokiomis prognozėmis galima pasitikėti.

Orų prognozių vartotojams lieka daryti prielaidą, kad prognozės nėra idealios: laikui bėgant jų tikslumas sparčiai mažėja. Mokslininkai teigia, kad nėra prasmės sudarinėti prognozių ilgesniam nei 10-ies dienų laikotarpiui, o šaltiniai, kurie laužo šią taisyklę pateikdami netikslas mėnesines prognozes, gali pakenkti viso meteorologijos mokslo reputacijai [27].

O ką apskritai reiškia tiksli orų prognozė? Jei mano gyvenamojoje vietoje konkrečiu laiku buvo prognozuotas lietus ir neiškrito nė lašo, bet krituliai pasirodė už kelių kilometrų, ar galima tokią prognozę laikyti visiškai netikslią?

Akivaizdu, kad gerai pasirinktas kritulių prognozių tikslumo matavimo būdas yra be galo svarbus norint atrasti geriausias orų prognozavimo metodus. Dar daugiau – norint įvairiapusiškai išanalizuoti gautus rezultatus, verta naudoti kelis tikslumo matavimo būdus.

Šiame poskyryje pristatomi trys skirtingi prognozių tikslumo įvertinimo metodai, kurie bus

naudojami darbe pristatomų algoritmų (žr. 5 skyrių) analizei:

- šaknis iš vidutinės kvadratinės paklaidos;
- HK įvertis;
- dalinis tikslumo įvertis.

#### 4.3.1 Šaknis iš vidutinės kvadratinės paklaidos

Šaknis iš vidutinės kvadratinės paklaidos – (angl. *root-mean-square error*, *RMSE*) Statistinis įvertis, naudojamas neatitikimui tarp spėjamo/išmatuoto rezultato ir žinomo faktinio įverčio matuoti. Plačiai naudojamas įvairių sričių statistinėje analizėje rezultatų tikslumui ir matavimo paklaidoms įvertinti. Šaknis iš vidutinės kvadratinės paklaidos apskaičiuojama pagal formulę (3):

$$RMSE = \sqrt{\sum_{i=1}^n \frac{(y'_i - y_i)^2}{n}} \quad (3)$$

čia:  $n$  – duomenų kiekis;  $y'_i$  – apskaičiuotos reikšmės;  $y_i$  – faktinės reikšmės.

RMSE naudojama ir orų prognozių tikslumui įvertinti [28, 29], nors tokiam matavimui ne visada gerai tinka, nes labai priklauso nuo matuojamo įvykio (mūsų atveju – kritulių) pasirodymo dažnio [30].

Be galutinio prognozių tikslumo įvertinimo mato, šis įvertis pasitelkiamas ir mūsų naudojamų dirbtinių neuroninių tinklų apmokymo proceso metu ir naudojamas kaip klaidos funkcija.

#### 4.3.2 HK įvertis

Ne visos klaidos yra vienodai reikšmingos: nepasitvirtinusi prognozė, kad žmogus serga kokia nors liga (klaidingai teigiama prognozė, angl. *false positive*, *FP*), yra ne tokia žalinga, nei testas, kuriam egzistuojančios ligos aptikti nepavyko (klaidingai neigiama prognozė, angl. *false negative*, *FN*).

Panašiai yra ir su orų prognozių tikslumo įvertinimu: du metodai gali turėti vienodas RMSE reikšmes, tačiau modelis, kuris tokį rezultatą pasiekia tiesiog spėliodamas, kad dažniausiai nelyja (statistiškai lyja rečiau nei nelyja), savo faktiniu tikslumu nusileis modeliui, kuris kartais daro klaidas prognozuojamas kritulių nebuvimą, tačiau gerai prognozuoja laikotarpius su krituliais.

**2 lentelė** Klaidų matrica

Stebėjimo reikšmė	Prognozuota reikšmė	
	Taip	Ne
Taip	TP	FN
Ne	FP	TN



Šias problemas sprendžia HK įvertis – (angl. *Hanssen-Kuiper score*) Klaidų matrica (angl. *confusion matrix*) besiremiantis prognozių tikslumo matas [31]. Klaidų matricoje (žr. 2 lentelę) apibrėžiami visi 4 galimi prognozių ir stebėjimų atitikimų/klaidų rezultatai:

- TP - (angl. *true positive*) – teisingai teigiamos prognozės (prognozuoti krituliai; krituliai fiksuoti);
- TN - (angl. *true negative*) – teisingai neigiamos prognozės (prognozuota be kritulių; krituliai nefiksuoti);
- FP - (angl. *false positive*) – klaidingai teigiamos prognozės (prognozuoti krituliai; krituliai nefiksuoti);
- FN - (angl. *false negative*) – klaidingai neigiamos prognozės (prognozuota be kritulių; krituliai fiksuoti).

HK įvertis duomenų rinkiniui apskaičiuojamas naudojant formulę (4) [31]:

$$HK = \frac{TP}{TP + FN} + \frac{TN}{FP + TN} - 1 \quad (4)$$

čia:  $TP, TN, FP, FN$  – aukščiau apibrėžiamų įvykių kiekiai viename duomenų rinkinyje. Mūsų atveju – vienoje prognozuojamo ir faktinio kritulių žemėlapiu poroje.

HK įvertis gali įgyti reikšmės nuo  $-1$  iki  $1$ , kur  $HK = 1$  – ideali prognozė,  $HK = 0$  – visiškai klaidinga, o  $HK = -1$  – visiškai priešinga stebėjimams.

HK įvertis literatūroje dar vadinamas tikrąja įgūdžio statistika (angl. *true skill statistics, TSS*) [32] ir yra plačiai naudojamas orų prognozių tikslumo matavimui [33, 34]. Įvertis išreiškia kaip gerai modeliai atskiria statistiškai teigiamus įvykius (lietus) nuo neigiamų (be lietaus).

Tiesa, HK tikslumo matas turi didelį trūkumą: įvesties duomenys gali įgyti tik dvi reikšmes: krituliai fiksuoti, arba ne. Galima įvertinti tik patį kritulių buvimo arba nebuvimo faktą, o ne kritulių stiprumo prognozės tikslumą.

### 4.3.3 Dalinis tikslumo įvertis

Aukščiau apibrėžti įverčiai apskaičiuojami sulyginant faktinį kritulių žemėlapi su prognoze: tas pačias koordinatas turintys pikseliai naudojami kaip duomenų pora apskaičiuojant ir RMSE, ir HK įverčius.

Naudojant šias metrikas erdviųjų (angl. *spatial*) prognozių tikslumui įvertinti, išryškėja viena problema: analizuojamos tik atitinkamų pikselių poros ir neįvertinama, kas vyksta gretimuose pikseliuose: galbūt prognozė buvo teisinga, tačiau prognozuoti krituliai iškrito už kelių kilometrų (poros pikselių) nuo tikrinamos teritorijos. RMSE, ir HK įverčiai tokiu atveju užfiksuotų maksimaliai klaidingą prognozę, nors faktiškai ji buvo teisinga bent iš dalies.

Šiai problemai išspręsti gali būti naudojamas dalinis tikslumo įvertis (angl. *fractions skill score, FSS*) FSS skirtas erdviųjų prognozių tikslumui matuoti: prognozės su stebėjimais lyginamos ne tik viename taške, tačiau ir to taško kaiminystėje (angl. *neighborhood*).

FSS apskaičiavimas vienai radaro žemėlapių porai yra sudėtingesnis, nei anksčiau aprašytų įverčių ir susideda iš keleto žingsnių. Visų pirma (kaip ir HK matmens atveju), abiejų paveikslėlių įvestys kiekviename taške konvertuojamos į dvejetaines reikšmes, pasirinkus norimą slenkstinę reikšmę  $q$  (5):

$$I_s = \begin{cases} 1, & \text{kai } S_r \geq q \\ 0, & \text{kai } S_r < q \end{cases} \quad I_p = \begin{cases} 1, & \text{kai } P_r \geq q \\ 0, & \text{kai } P_r < q \end{cases} \quad (5)$$

čia:  $S_r$  ir  $P_r$  – stebėta ir prognozuojama kritulių stiprumo kategorija taške;  $I_s$  ir  $I_p$  – apskaičiuojama dvejetainė stebėjimo ir kritulių prognozės reikšmė. Mūsų atveju,  $q = 1$ .

Toliau išskiriame  $n$  dydžio dalis (apylinkes) kiekviename paveikslėlio taške  $(i, j)$  (6, 7):

$$S(n)(i, j) = \frac{1}{n^2} \sum_{k=1}^n \sum_{l=1}^n I_s \left[ i + k - 1 - \frac{n-1}{2}, j + l - 1 - \frac{n-1}{2} \right] \quad (6)$$

$$P(n)(i, j) = \frac{1}{n^2} \sum_{k=1}^n \sum_{l=1}^n I_p \left[ i + k - 1 - \frac{n-1}{2}, j + l - 1 - \frac{n-1}{2} \right] \quad (7)$$

čia:  $I_s[\dots, \dots]$  ir  $I_p[\dots, \dots]$  – stebima ir prognozuojama dvejetainė kritulių reikšmė apskaičiuojamose koordinatėse (taškams už matricos ribų priskiriama reikšmė 0);  $n$  – pasirinktas apylinkės dydis;  $S(n)(i, j)$  ir  $P(n)(i, j)$  – apskaičiuojami daliniai tikslumo įverčiai taške  $(i, j)$ .

Sekantis žingsnis yra apskaičiuoti vidutinę kvadratinę paklaidą MSE, naudojant gautus dalinius tikslumo įverčius  $S(n)$  ir  $P(n)$  (8):

$$MSE_{(n)} = \frac{\sum_{i=1}^{N_x} \sum_{j=1}^{N_y} (S(n)_{i,j} - P(n)_{i,j})^2}{N_x N_y} \quad (8)$$

čia:  $S(n)(i, j)$  ir  $P(n)(i, j)$  – apskaičiuoti daliniai tikslumo įverčiai;  $N_x, N_y$  – analizuojamos teritorijos matmenys, mūsų atveju,  $N_x = N_y = 64$ ;  $MSE_{(n)}$  – vidutinė kvadratinė paklaida visose  $n$  dydžio apylinkėse.

Norėdami apskaičiuoti FSS, apibrėžiame referencinę vidutinės kvadratinės paklaidos vertę, kurią galima įsivaizduoti, kaip didžiausią galimą MSE apskaičiuotuose daliniuose įverčiuose (9):

$$MSE_{(n)ref} = \frac{\sum_{i=1}^{N_x} \sum_{j=1}^{N_y} S(n)_{i,j}^2 + P(n)_{i,j}^2}{N_x N_y} \quad (9)$$

Galiausiai, FSS apskaičiuojamas naudojantis formule (10):

$$FSS_{(n)} = 1 - \frac{MSE_{(n)}}{MSE_{(n)ref}} \quad (10)$$

Mūsų atveju, tikslumo matavimui pasirenkame  $n = 4$ . Įvertis gali įgyti reikšmes nuo 0 iki 1, kur 0 atitinka visiškai klaidingą, o 1 – idealią prognozę. FSS įverčio apskaičiavimo žingsniai ir formulės parengti pagal [30].

Nors ir gerai tinka erdvinių prognozių įvertinimui, FSS, kaip ir ankstesni metodai turi porą trūkumų:

- kaip ir HK įvertis, tinka tik dvejetainėms įvestims įvertinti;
- dėl didelio algoritmų žingsnių kiekio, FSS įverčio apskaičiavimas užtrunka ilgiau nei RMSE ir HK metrikų.

#### 4.3.4 Prognozių tikslumo matavimo metodų apibendrinimas

Visi trys pristatyti metodai RMSE, HK ir FSS į vertinamas prognozes leidžia pažvelgti iš skirtingų aspektų:

- RMSE leidžia įvertinti kritulių stiprumo prognozavimo tikslumą;
- HK įvertis išreiškia teisingai teigiamų kritulių prognozių svarbą prieš prognozes, kad krituliai neiškris;
- FSS leidžia analizuoti prognozes erdviniu aspektu.

Kaip jau minėta, HK ir FSS įverčiai palaiko tik įvestis su dvejomis loginėmis reikšmėmis. Ši trūkumą buvo bandyta spręsti apskaičiuojant kelias metrikų reikšmes su skirtingu slenkstiniu įverčiu  $q$ , ir suvidurkinant gautus rezultatus. Tiesa, šis sprendimas rezultatų iš esmės nepakeitė: keitėsi tik absoliutinės metrikų reikšmės, santykis tarp skirtingų algoritmų rezultatų išlikdavo toks pat. Be to, pakartotinis įverčių skaičiavimas smarkiai pailgindavo rezultatų tikslumo tyrimo trukmę. Dėl šios priežasties buvo nuspręsta naudoti originalų įverčių apskaičiavimo būdą.

Eksperimentinėje dalyje (žr. 5 skyrių) bus naudojami visi trys aptarti metodai. Tai leis visapusiškai ištirti kritulių prognozavimo modelius ir įgyti dar daugiau svarbių įžvalgų apie gautus rezultatus.

## 5 Eksperimentas

### 5.1 Eksperimento sąlygos

Šiame poskyryje trumpai aprašomas visų tyrimo dalyje naudojamų neuroninių tinklų modelio apmokymo procesas, kad eksperimento sąlygos galėtų būti atkartojamos.

#### 5.1.1 Modelių apmokymo metodas

1. Analizuojamai tinklo architektūrai pasirenkami pradiniai modelio hiperparametrai: filtrų skaičius ir filtrų branduolio dydis (angl. *kernel size*);
2. Modelis apmokomas naudojant praėjusiame skyriuje aprašytus duomenis (žr. 5.1.2 skirsni);
3. Po kiekvienos epochos modelis validuojamas: algoritmas generuoja prognozes 8-iems žingsniams į priekį. Naudojantis FSS prognozių tikslumo įvertinimo metodika, apskaičiuojamas prognozių atitikimas faktinėms sąlygoms, randamas šios metrikos vidurkis.
4. Jei gaunamas metrikos vidurkis didesnis už iki tol geriausią, dabartiniai tinklo parametrai išsaugojami kaip geriausi;
5. Žingsniai 2–4 kartojami **50** apmokymo epochų ir gaunamas algoritmo vidinių parametru rinkinys, geriausiai prognozuojantis kritulius pagal validavimo duomenis.

#### 5.1.2 Modelio apmokymo parametrai

Algoritmų apmokymui naudojami šie statiniai parametrai:

- duomenų egzemplioriai vienam apmokymo ciklui (angl. *batch size*): **35**;
- apmokymo greitis (angl. *learning rate*): **0,005**, tolygiai mažėjantis su kiekviena epocha  $2 \times 10^{-6}$  greičiu; apmokymo greitis gali būti nežymiai koreguojamas, jei algoritmas įstringa lokaliame minimume;
- neuroninio tinklo stochastinio optimizavimo algoritmas: **Adam'o** [35];
- klaidos funkcija: **MSE**;
- visų sluoksnių neuronų aktyvavimo funkcija: **ReLU** (angl. *rectified linear unit*) (11).

$$ReLU(x) = \begin{cases} x, & \text{kai } x \geq 0 \\ 0, & \text{kai } x < 0 \end{cases} \quad (11)$$

#### 5.1.3 Prognozių generavimo ir jų validavimo procesas

Visos 5.3 skyriuje pristatomos neuroninių tinklų architektūros išvestyje generuoja tik vieną prognozės žemėlapi, atitinkantį prognozuojamas oro sąlygas už 15-os minučių. Norint gauti

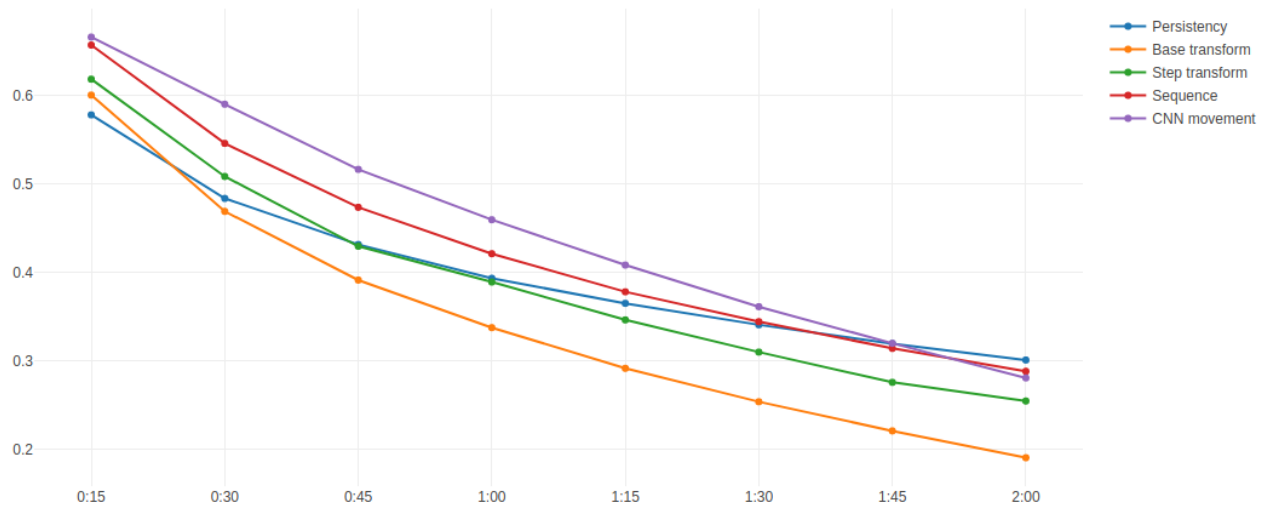
didesnės trukmės prognozę, neuroninio tinklo vykdymo procesas kartojamas norimą kiekį kartų, išvesties paveikslėlių prijungiant prie sekančio prognozės žingsnio įvesties paveikslėlių (12, 13):

$$X_1 = [x_{n-3}, x_{n-2}, x_{n-1}, x_n]; \quad Y_1 = CNN(X_1) \quad (12)$$

$$X_2 = [x_{n-2}, x_{n-1}, x_n, Y_1]; \quad Y_2 = CNN(X_2) \quad (13)$$

Čia:  $x_{n-3}, x_{n-2}, x_{n-1}, x_n$  – nuosekli radaro paveikslėlių seka prognozės momentu  $n$  ir prieš tai sekusiais laiko atskaitos taškais  $n-1, n-2, n-3$ ;  $X_1, X_2$  – iš įvesties paveikslėlių formuojama neuroninio tinklo įvestis.  $Y_1, Y_2$  – pirmosios ir antrosios iteracijos neuroninio tinklo išvestys;  $CNN$  – apmokyto neuroninio tinklo algoritmas.

Kadangi prognozių tikslumo įvertinimas dideliame kiekiu egzempliorių yra daug resursų ir laiko reikalaujantis procesas (o lietaus nebuvimo prognozavimas kai nelyja yra trivialus), visų eksperimentų metu modelių tikslumo įvertinimas bus atliekamas tik su radaro žemėlapiais, kuriuose kritulių padengimas yra  $\geq 4\%$ .



**28 pav.** Ankstyvųjų eksperimentų su deterministiniais modeliais rezultatai, prognozių tikslumui įvertinti naudojamas HK įvertis; geriausius rezultatus parodė konvoliucinis neuroninis tinklas (CNN movement)

## 5.2 Eksperimentai su deterministiniais modeliais

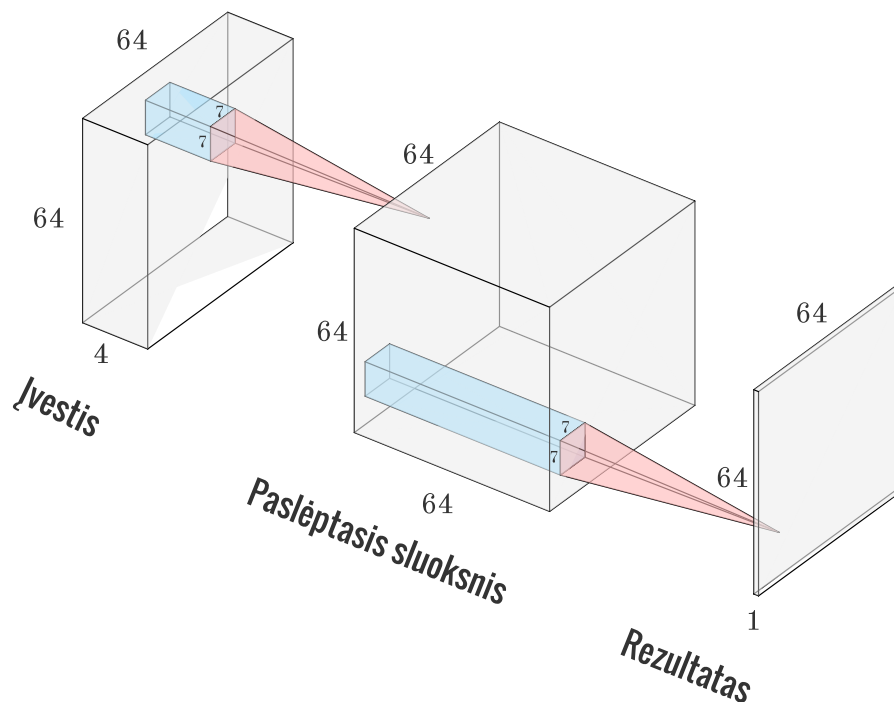
Ankstyvajame magistro projekto tyrimo etape buvo atliktas deterministinių (nenaudojančių mašininio mokymosi) orų prognozių algoritmų modelių tyrimas (žr. 4 priedą) ir parašytas IVUS konferencijoje 2018-aisiais metais pristatytas straipsnis. Jame analizuojamas trumpalaikis kritulių prognozavimas, naudojant nesudėtingus algoritmus: rezultatai generuojami „perkeliant“ viso žemėlapio atvaizdą surasto kritulių judėjimo vektorius kryptimi. Straipsnyje pristatomas ir 3-jų sluoksnių konvoliucinio neuroninio tinklo modelis (5.3.1 skirsnyje aprašomo modelio pirmtakas).

Gauti rezultatai parodė aiškius konvoliucinio neuroninio tinklo modelio pranašumus prieš sąlygų išsilaikymo įvertį (angl. *persistence*) ir kitus tirtus deterministinius modelius (žr. 28 pav.). Dėl šios priežasties, magistro projekte analizuojami tik neuroniniais tinklais paremti kritulių prognozavimo algoritmai.

## 5.3 Prognozės modelių architektūros, besiremiančios neuroniniais tinklais

Šiame poskyryje aprašomos eksperimente naudojamų neuroninių tinklų modelių architektūros ir ypatybės.

### 5.3.1 3-jų sluoksnių CNN



**29 pav.** 3-jų sluoksnių konvoliucinis neuroninis tinklas su 4 kanalais įvestyje, 64 filtrais paslėptajame sluoksnyje ir nekintančiu 7 x 7 filtro branduolio dydžiu tarp visų trijų sluoksnių

3-jų sluoksnių neuroninis tinklas – paprasčiausias dirbtinio konvoliucinio neuroninio tinklo variantas, sudarytas iš trijų sluoksnių: įvesties, paslėptojo ir išvesties. Kadangi įvestis ir išvestis turi tas pačias pločio ir aukščio dimensijas, tinkle nėra daugeliui konvoliucinių neuroninių tinklų įprastų sutelkimo imant maksimalią reikšmę sluoksnių (angl. *max pooling*).

Tinklo sluoksniuose naudojamas filtro branduolio padidinimas (angl. *padding*): branduolio įvestis padidinama tuščių (nulinių) reikšmių apvadu, kad tarp atskirų sluoksnių būtų išsaugojamos paveikslėlio aukščio ir pločio dimensijos.

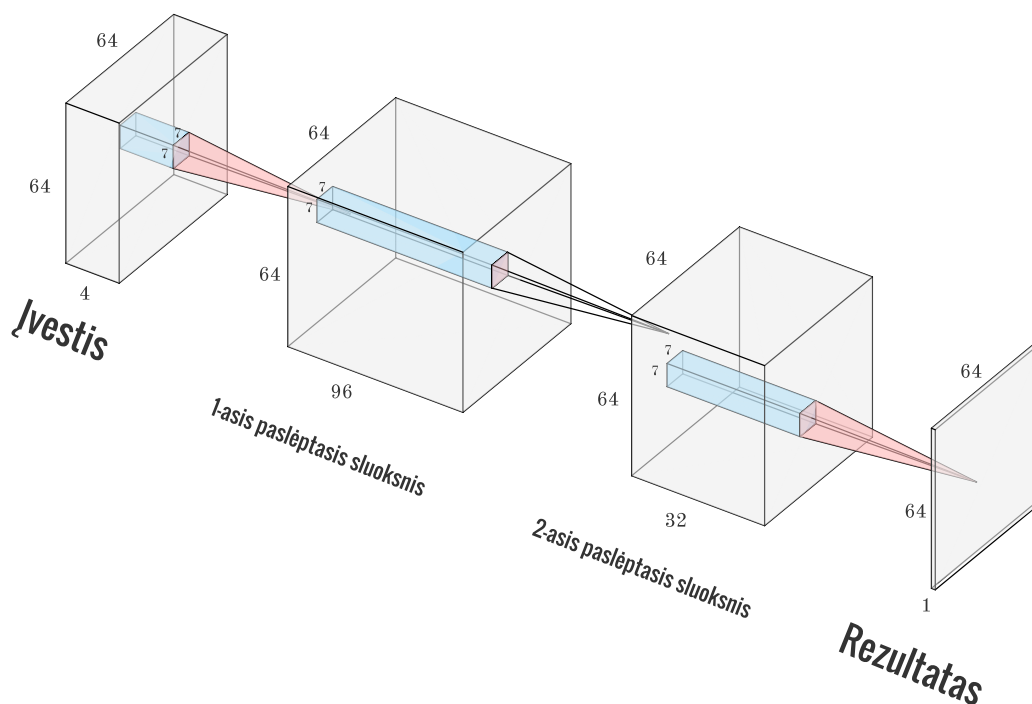
Neuroninio tinklo įvestį sudaro 4 kanalai, susidedantys iš 4-ių iš eilės einančių radaro paveikslėlių. Jei įvestyje pridedama atsitiktinai sugeneruota ar faktinė vietovės reljefo informacija (žr. 5.4 skyrių), įvestį gali sudaryti 5 ir daugiau kanalų.

Trijų sluoksnių konvoliucinio neuroninio tinklo iliustracija pateikiama 29 pav.

### 5.3.2 4-ių sluoksnių CNN

4-ių sluoksnių konvoliucinis tinklas atitinka 3-jų sluoksnių tinklą su papildomu paslėptuoju sluoksniu. Tokia architektūra tinklui potencialiai leidžia išmokti didesnę informacijos kiekį. Didėja ir galimų hiperparametrų kiekis: galima nustatyti tris skirtingus filtro branduolio dydžius ir du filtrų kiekius paslėptuosiuose sluoksniuose.

Šio tipo dirbtinio neuroninio tinklo architektūros pavyzdys pateikiamas 30 pav.



**30 pav.** 4-ių sluoksnių konvoliucinis neuroninis tinklas su 4 kalanalais įvestyje, 96 filtrais 1-ajame paslėptajame sluoksnyje; 32 filtrais 2-ajame paslėptajame sluoksnyje ir nekintančiu 7 x 7 filtro branduolio dydžiu tarp visų trijų sluoksnių

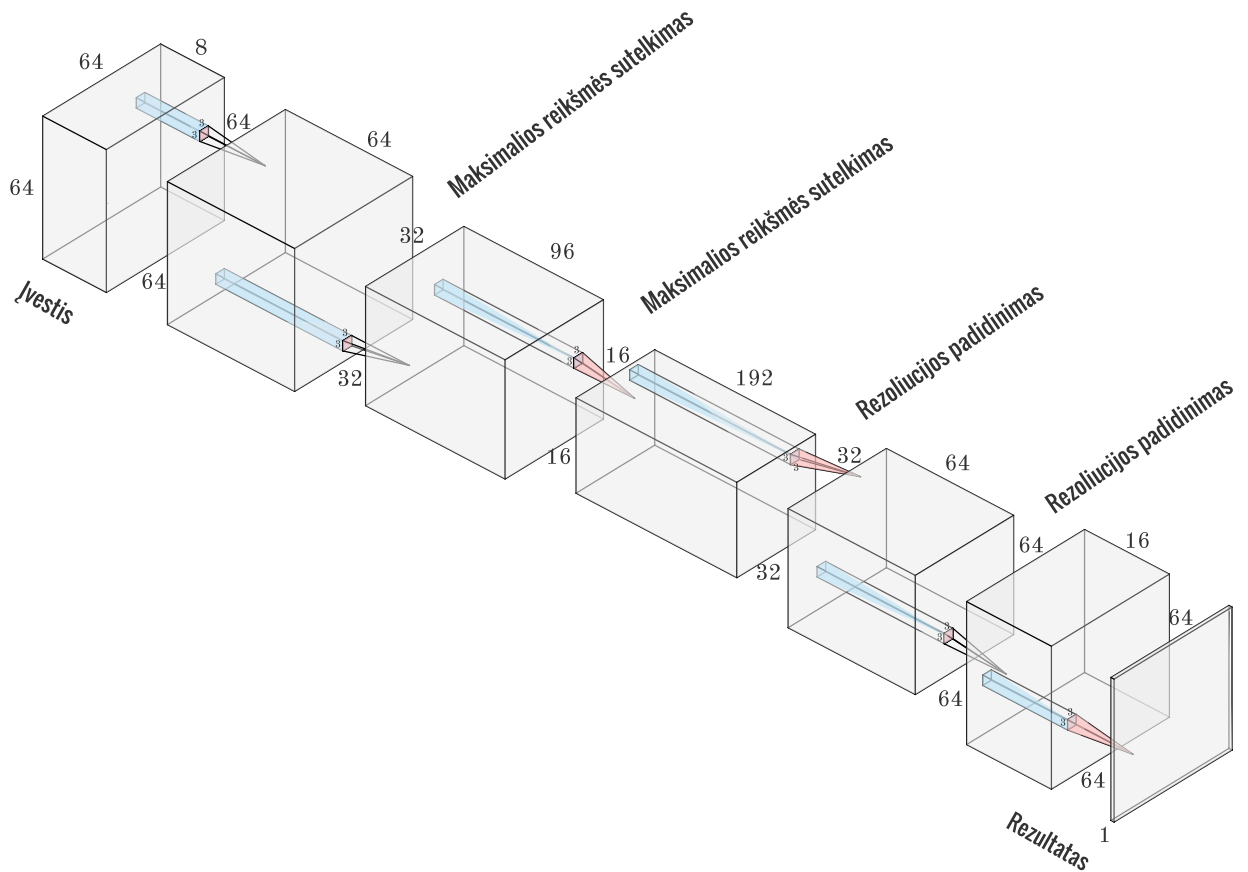
### 5.3.3 Autoenkoderio architektūra

Autoenkoderio neuroninis tinklas susideda iš dviejų dalių: enkoderio, sumažinančio įvesties dimensiją iki tam tikros kompaktiškos reprezentacijos ir dekoderio, kuris, pasinaudodamas šia reprezentacija atkuria išvestį su originaliomis dimensijomis.

Šiame projekte pristatoma autoenkoderio struktūra naudoja konvoliucinius sluoksnius bei rezoliuciją mažinančią maksimalios reikšmės sutelkimo (angl. *max. pooling*) operaciją. Originalus dydis gaunamas pasinaudojus sutelkimui priešingu rezoliuciją didinančiu (angl. *upsampling*) sluoksniu.

Šios architektūros tikslas yra enkoderio dalyje „išmokti“ užkoduoti svarbiausias prognozei ir vaizdo atkūrimui reikalingas įvesties savybes, t.y. atlikti savybių išskyrimą (angl. *feature extraction*). Dekoderis savo ruožtu siekia enkoderio sugeneruotą tarpinę reprezentaciją panaudoti reikiamai modelio išvesčiai apskaičiuoti.

Šiame tyrime bus naudojami autoenkoderiai su viena arba dvejomis sutelkimo ir rezoliucijos didinimo operacijų poromis ir konvoliuciniais sluoksniais tarp jų. Autoenkoderio architektūros pavyzdys pateikiamas 31 pav.



**31 pav.** Autoenkoderio tipo konvoliucinis neuroninis tinklas su dvejomis sutelkimo ir rezoliucijos padidinimo operacijų poromis, kintančiu filtrų kiekiu sluoksniuose (64, 96, 192, 64, 16) ir statiniu  $3 \times 3$  branduolio filtro dydžiu.

### 5.3.4 *Conv-LSTM* architektūra

Pirmą kartą šio tipo neuroninis tinklas pristatytas būtent kritulių prognozavimui pagal kritulių radaro žemėlapius [36].

Projekte naudojamas *Conv-LSTM* tinklas savo išorine struktūra panašus į bet kurią kitą konvoliucinį tinklą, tačiau šio tipo tinklo sluoksniuose naudojama kitokia neuronų struktūra, įvestyje turinti ne tik dabartinio apdorojamo sluoksnio duomenis, bet ir informaciją apie ankstesnių įvesčių rezultatus (žr. 32).

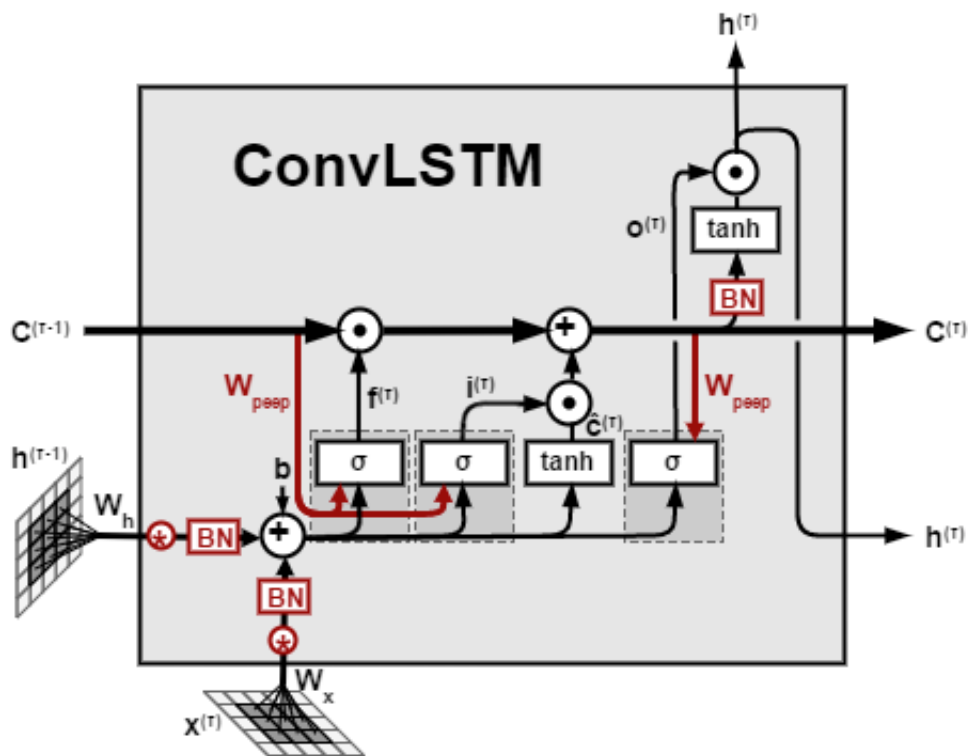
## 5.4 Papildomo reljefo ir atsitiktinio duomenų kanalų efektyvumo tyrimas 3-jų sluoksnių konvoliuciniame neuroniniame tinkle

### 5.4.1 Eksperimento aprašymas

Konvoliucinio neuroninio tinklo filtro branduoliai viename sluoksnyje dalijasi tais pačiais svoriais. Tai reiškia, kad nepriklausomai nuo to, kurioje geografinėje vietoje bus matomi krituliai, konvoliucinis sluoksnis juos apdoros identišškai. Tai gali mažinti prognozavimo algoritmų tikslumą, nes virš vandens ir žemės krituliai elgiasi skirtingai, o vietovėje iškritęs kritulių kiekis priklauso ir nuo teritorijos reljefo (žr. 2.2.1 skyrių).

Šią problemą potencialiai galima išspręsti prie įvesties žemėlapių prijungus papildomus sta-





**32 pav.** Konvoliucinio LSTM tinklo sluoksnio architektūra. Sluoksnis panašus į paprastą LSTM sluoksnį, tačiau šiuo atveju pilnai sujungtų matricių dauginimo operacijos pakeičiamos konvoliucinėmis operacijomis. Šaltinis: [medium.com](https://medium.com)

tinus kanalus: paveikslėlį su atsitiktinėmis pikselių reikšmėmis (žr. 33 a pav.), ar vietovės reljefo žemėlapi (žr. 33 b pav.) [37].

Pilnai išanalizuoti visus pristatomas neuroninio tinklo architektūrų, hiperparametrų ir papildomų kanalas kombinacijas yra pernelyg didelės apimties darbas. Dėl šios priežasties, pirmiausia reikia nuspręsti, ar papildomi kanalas tikrai padidina kritulių prognozių tikslumą, ir jei taip, kuris iš dviejų analizuojamų kanalas tam tinka geriau (o galbūt verta panaudoti jų kombinaciją).

Šiam tyrimui bus naudojamas paprasčiausias 3-jų sluoksnių konvoliucinis neuroninis tinklas, validavimui skirtų duomenų (**1 760 egzempliorių**) aibė ir lengviausiai apskaičiuojama RMSE paklaida.

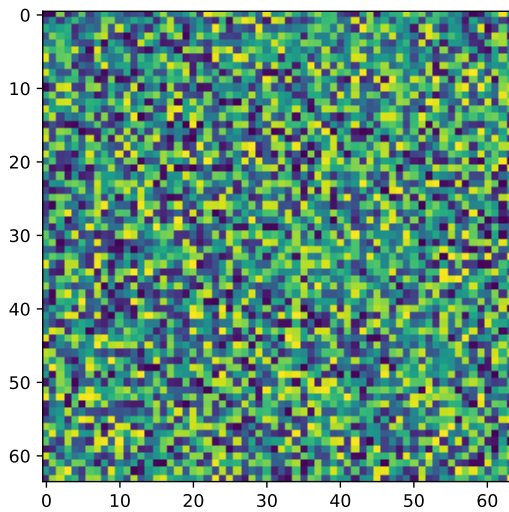
Eksperimentų rezultatai lyginami su baziniu sąlygų išsilaikymo (angl. *persistence*) „algoritmu“, kuris visiems prognozės laiko žingsniams grąžina tą patį bazinio prognozės generavimo laiko paveikslėlį (14):

$$X_1 = [x_{n-3}, x_{n-2}, x_{n-1}, x_n]; \quad Y_1, Y_2, \dots, Y_N = x_n \quad (14)$$

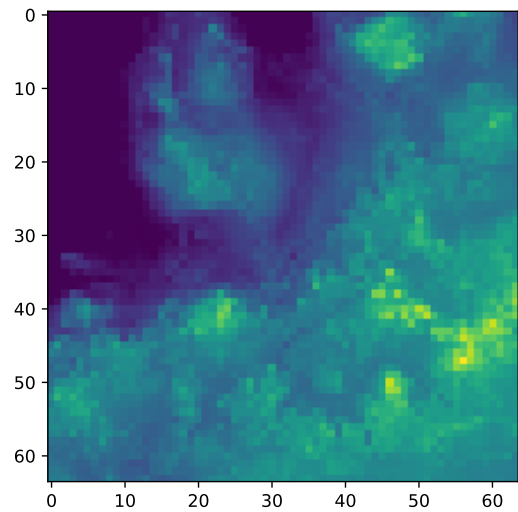
#### 5.4.2 Eksperimento eiga

Eksperimente bus naudojami šie trijų sluoksnių modelių parametru rinkiniai:

- 64 filtrai paslėptajame sluoksnyje ir  $3 \times 3$  filtro branduolio dydis (poskyrio diagramose naudojamas formatas: 64\_3x3);



(a) Statinis atsitiktinių reikšmių kanalas



(b) Vietovės reljefo duomenys (aukštis virš jūros lygio) analizuojamoje teritorijoje

**33 pav.** Atsitiktinių reikšmių (a) ir vietovės reljefo kanalas (b) analizuojamoje teritorijoje. Reljefo duomenų šaltinis: <https://elevation-api.io/>

- 72 filtrai,  $8 \times 8$  dydis (72\_8x8);
- 82 filtrai,  $7 \times 7$  dydis (82\_8x8);
- 96 filtrai,  $3 \times 3$  dydis (92\_3x3).

Kiekvieną parametų rinkinį naudoja konvoliuciniai neuroniniai tinklai su skirtingais papildomais kanalais:

- be papildomo kanalo (poskyrio diagramose naudojamas be papildomo prefikso);
- su atsitiktinių reikšmių papildomu kanalu (*rand*);
- su vietovės reljefo duomenimis papildomame kanale (*elev*).

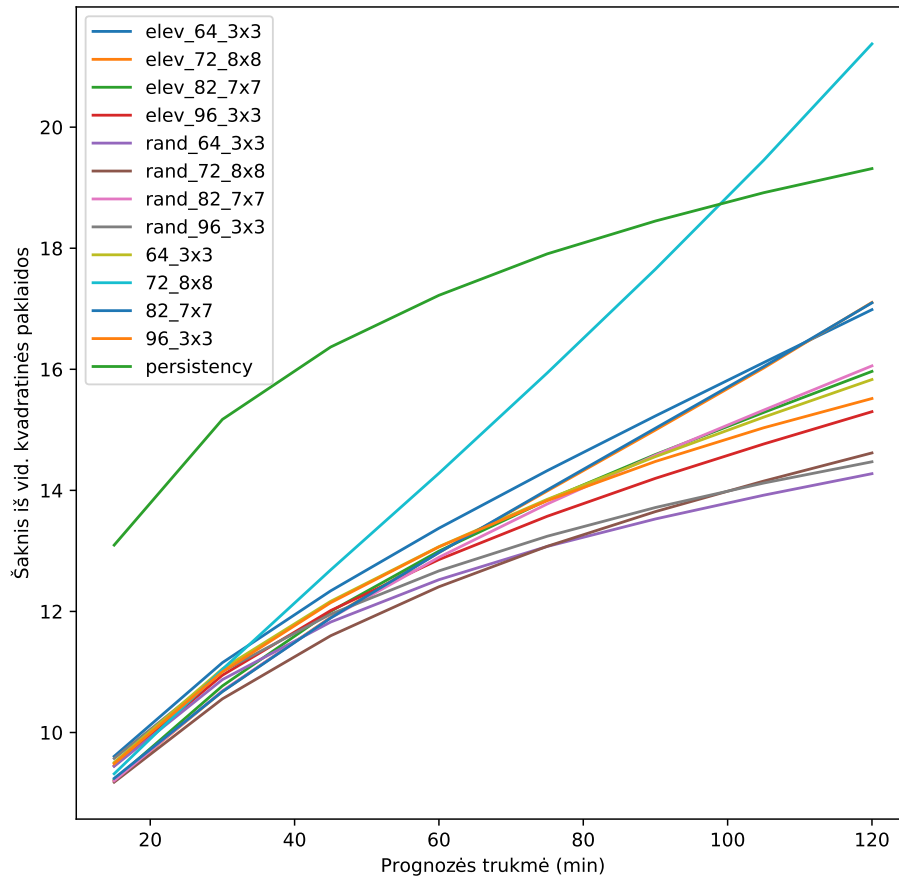
Iš viso gaunama **12** skirtingų tuo pačiu metodu apmokytų trijų sluoksnių konvoliucinių neuroninių tinklų.

### 5.4.3 Eksperimento rezultatai

Visų gautų modelių validavimo rezultatai (remiantis RMSE įverčiu) pateikiami 34 pav.

Beveik visi modeliai aplenkė bazinių sąlygų išsilaikymo (angl. *persistence*) įvertį: daugumos modelių RMSE buvo apie trečdaliu mažesnis, nei šios bazinės metrikos. Tai – teigiamas rezultatas, nes modelių tikslumas gerokai viršija bazinių sąlygų išsilaikymo (angl. *persistence*) tikslumą.

Iš visų tirtų mašininio mokymosi modelių geriausiai pasirodė CNN, turintys papildomą kanalą iš atsitiktinai sugeneruotų pikselių reikšmių. Siekiant išskirti atskirus skirtingų papildomų kanalų tinklų rezultatus, pateikiama dar viena diagrama, kurioje pavaizduoti geriausi kiekvienos papildomo kanalo rūšies rezultatai (žr. 35 pav.). Geriausi modeliai pasirinkti pagal kiekvienos kreivės RMSE skirtingų prognozės trukmių klaidos vidurkį.



**34 pav.** Visų trijų sluoksnių konvoliucinių neuroninių tinklų *RMSE* palyginimas remiantis *RMSE* metrika. Žalios spalvos kreivė viršuje – bazinio išsilaikymo įvertis, kurį beveik visiems modeliams pavyko aplenkti.

Geriausių modelių *RMSE* yra panašūs, tačiau geriausią rezultatą pasiekė CNN su papildomu atsitiktinių reikšmių kanalu,  $8 \times 8$  branduolio dydžiu ir 72 filtrais paslėptajame sluoksnyje (rand\_72\_8x8). Antrą geriausią rezultatą parodė CNN su papildomu reljefo kanalu (elev\_96\_3x3), nuo jo nežymiai atsiliko CNN be papildomo kanalo (96\_3x3).

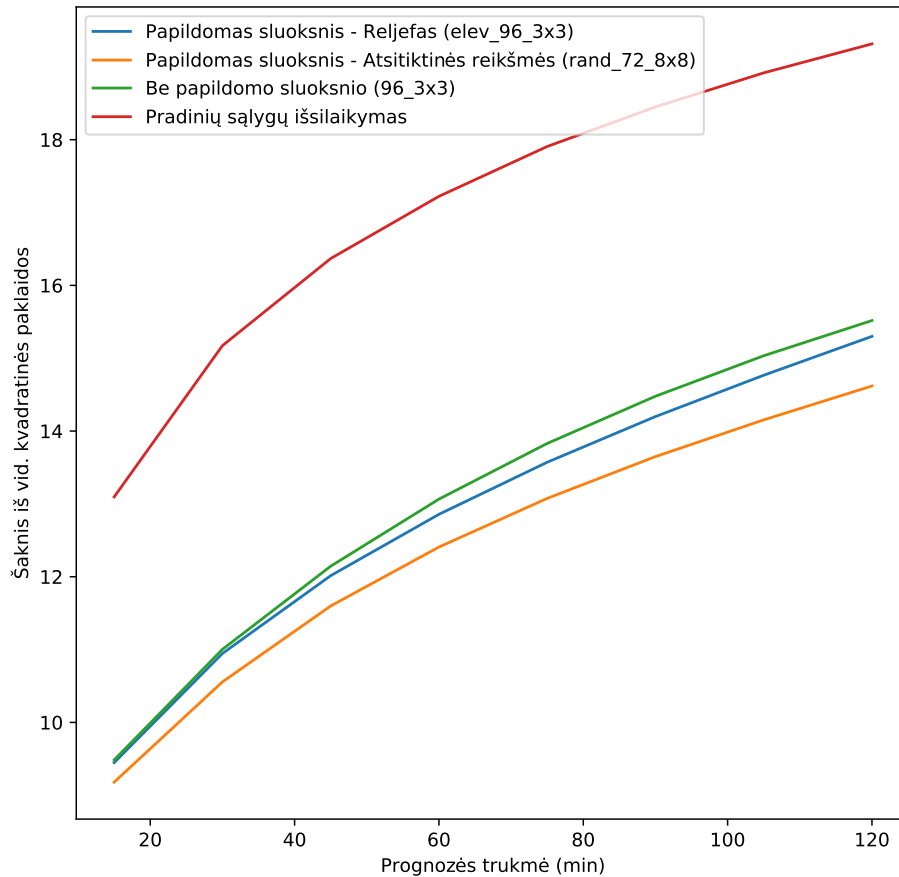
Šių trijų geriausių modelių atvejų analizė, kur keliems konkreitiems kritulių judėjimo atvejams pateikiami šių modelių sugeneruotų prognozių pavyzdžiai, pateikiama 1-ame priede.

3 lentelėje pateikiami kiekvienos papildomo kanalo rūšies tinklų vidutiniai *RMSE* įverčiai:

**3 lentelė** Trijų sluoksnių CNN modelių su skirtingais papildomais kanalais *RMSE* palyginimo lentelė

Modelis	RMSE
persistency	17,056
96_3x3	13,0694
elev_96_3x3	12,8877
rand_72_8x8	12,4047

Panašu, kad hipotezė, jog papildomas statinis duomenų kanalas leis modeliui pasiekti geresnių rezultatų, pasiteisino, tiesa, nežymiai. Dar vienas įdomus pastebėjimas: CNN su atsitiktinių



**35 pav.** Geriausių kiekvienos rūšies trijų sluoksnių konvoliucinių neuroninių tinklų palyginimas remiantis RMSE metrika. Bazinis sąlygų išsilaikymo įvertis pateikiamas raudona spalva.

duomenų kanalu pasiekė geresnių rezultatų nei su papildomu reljefo duomenų kanalu, nors, atrodytų, pastarieji duomenys tinklui turėtų suteikti daugiau informacijos apie kritulių judėjimo vietovėje ypatumus.

Gali būti, kad kontrastingi atsitiktiniame duomenų kanale esantys taškai (žr. 33 a pav.) tinklui leidžia lengviau atpažinti analizuojamos zonos koordinatas, nei pakankamai vienalytis reljefo kanalas (žr. 33 b pav.). Tiesa, toks ryškus kaimyninių pikselių skirtumas gali sukelti tinklo prognozių nestabilumą: ypač ryškūs atsitiktinio kanalo pikseliai tam tikrais atvejais gali sąlygoti nekontroliuojamai išaugusias išvesties reikšmes. Akivaizdu, kad būtų naudinga atlikti detalesnį tyrimą.

Modelių sugeneruotų prognozių pavyzdžius galima rasti 1-ajame priede.

Kitame poskyryje (žr. 5.5 poskyrį) nagrinėjamos įvairios atsitiktinių duomenų kanalų variacijos ir jų įtaka kritulių prognozavimo tikslumui.

## 5.5 Papildomų reljefo ir atsitiktinio duomenų kanalų kombinacijų tyrimas

### 5.5.1 Eksperimento aprašymas

Ankstesniame poskyryje atlikto eksperimento (žr. 5.4 poskyrį) rezultatai parodė, kad CNN prognozavimo tikslumas didėja įvestyje pridėjus pikselių koordinates leidžiančius nustatyti statinių duomenų kanalus. Galbūt dar geresnį rezultatą būtų galima pasiekti panaudojus įvairias šių papildomų kanalų kombinacijas.

### 5.5.2 Eksperimento eiga

Visiems šiame poskyryje vykdomame tyrime naudojamiems neuroniniams tinklams taikoma ta pati architektūra: 96 filtrai paslėptajame sluoksnyje ir  $3 \times 3$  filtro branduolio dydis. Su tokiais parametrais geriausią rezultatą pasiekė CNN su papildomu reljefo kanalu ir be papildomo kanalo. Tyrimui naudojamos šeši papildomų sluoksnių deriniai:

- 1 atsitiktinių duomenų kanalas, 0 reljefo kanalas duomenų;
- 1 atsitiktinių duomenų kanalas, 1 reljefo kanalas duomenų;
- 2 atsitiktinių duomenų kanalas, 0 reljefo kanalas duomenų;
- 2 atsitiktinių duomenų kanalas, 1 reljefo kanalas duomenų;
- 3 atsitiktinių duomenų kanalas, 0 reljefo kanalas duomenų;
- 3 atsitiktinių duomenų kanalas, 1 reljefo kanalas duomenų.

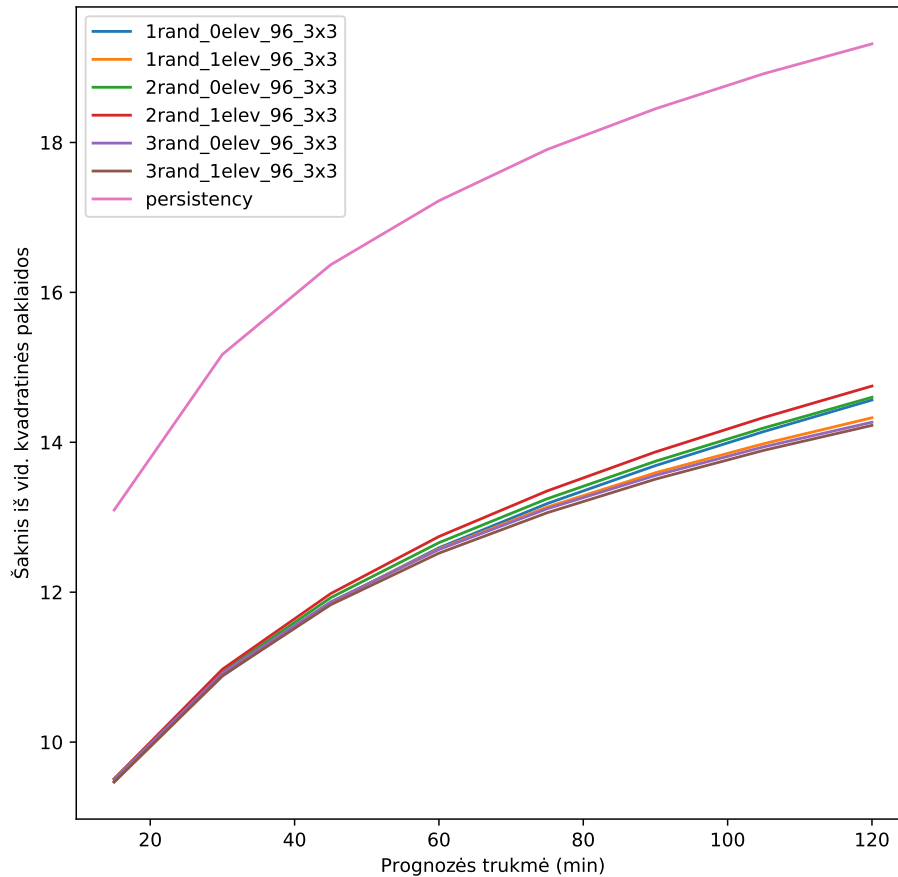
Modelių pavadinimai generuojami pagal formulę:  $Xrand\_Yelev\_96\_3x3$ , kur  $X$  – papildomų atsitiktinių duomenų kanalų skaičius, o  $Y$  – papildomų reljefo duomenų kanalų skaičius. Modeliai validuoti tais pačiais duomenimis kaip ir 5.4 poskyryje aprašomame eksperimente.

### 5.5.3 Eksperimento rezultatai

Rezultatai pateikiami 36 pav. ir 4 lentelėje.

**4 lentelė** Trijų sluoksnių CNN su įvairiais papildomų kanalų deriniais vidutinės RMSE palyginimas

Modelis	RMSE
persistency	17,056
2rand_1elev_96_3x3	12,6883
2rand_0elev_96_3x3	12,6021
1rand_0elev_96_3x3	12,5475
1rand_1elev_96_3x3	12,4791
3rand_0elev_96_3x3	12,4678
3rand_1elev_96_3x3	12,424



**36 pav.** Trijų sluoksnių CNN su įvairiais papildomų kanalų deriniais vidutinės RMSE palyginimas. Bazinis sąlygų išsilaikymo įvertis – viršutinė rožinė kreivė.

Iš rezultatų matome, kad visi analizuoti modeliai pasirodė labai panašiai: skirtumas tarp geriausio ir prasčiausio modelio RMSE buvo **0,26**. Geriausiai pasirodė CNN, turėjęs didžiausią bendrą papildomų statinių kanalų skaičių.

Nors skirtumas ir nežymus, net ir nedidelis papildomų kanalų panašumas gali sąlygoti didesnę modelių tikslumą didėjant analizuojamos architektūros sudėtingumui.

Eksperimento rezultatai rodo, kad modelių įvestyse tikslinga naudoti bent tris statinius atsitiktinių duomenų kanalus ir vieną statinį duomenų kanalą su vietovės reljefo duomenimis. Tokio tipo duomenų įvesčių konfigūracija bus naudojama ir paskutiniame eksperimente.

## 5.6 Išsamus skirtingų architektūrų neuroninių tinklų prognozių generavimo tyrimas

### 5.6.1 Eksperimento aprašymas

Pasinaudojus per pirmuosius du eksperimentus sukaupta patirtimi, metas ištirti sudėtingesnių modelių tikslumą. Bus tiriami keturių skirtingų architektūrų modeliai: trijų sluoksnių CNN, keturių sluoksnių CNN, autoenkoderis ir konvoliucinis LSTM. Visos šios architektūros detalios aprašytos 5.3 poskyryje. Kiekviena architektūra išbandoma su 2-3 skirtingais hiperparametru (filtrų kiekio) rinkiniais.

Siekiant sumažinti skirtingų parametrų rinkinių skaičių, visų modelių visuose sluoksniuose naudojamas  $3 \times 3$  filtrų branduolio dydis – didžioji dalis 5.4 skyriuje aprašomų modelių su šiuo filtrų branduoliu pasiekė geriausius rezultatus.

Šis eksperimentas remiasi visiškai atskiru testavimo duomenų rinkiniu. Siekiant išvengti bet kokios tiesioginės ar netiesioginės įtakos modelių apmokymo sprendimams, šie duomenys nebuvo niekaip analizuojami ar naudojami ankstesniuose eksperimentuose.

### 5.6.2 Eksperimento eiga

Visos naudojamos architektūros vienaip ar kitaip remiasi konvoliuciniais sluoksniais, tačiau skiriasi savo dydžiu ir apmokymo sudėtingumu. 5 lentelėje pateikiama analizuojamų architektūrų vidutinė vienos epochos apmokymo trukmė. Panašiai skiriasi ir jau apmokyto tinklo vykdymo trukmė, tačiau laiko skirtumas juntamas mažiau, nes vienu metu apdorojamas žymiai mažesnis duomenų kiekis.

**5 lentelė** Įvairių neuroninių tinklų architektūrų apmokymo trukmė ir apmokomų parametrų kiekis

Architektūra	1 Epochos trukmė	Žingsnio trukmė	Apm. par. kiekis <sup>a</sup>
3-jų sluoksnių CNN (82 filtrai)	17 s	562 $\mu$ s	7,873
4-ių sluoksnių CNN (64, 96 filtrai)	44 s	2 ms	62,945
Autoenkoderis (5 sluoksniai)	60 s	2.5 ms	71,169
Autoenkoderis (7 sluoksniai)	80 s	3 ms	355,393
Konvoliucinis LSTM (96, 64 filtrai)	161 s	6 ms	364,481

<sup>a</sup>Apmokomų parametrų kiekis

Visi apmokyti modeliai buvo ištestuoti su tuo pačiu duomenų rinkiniu, atfiltruotu pagal minimalią 4% lietaus padengimo ribą.

Eksperimento rezultatai pateikiami tokia tvarka: pirmiausia atliekamas kiekvienos analizuojamos architektūros sugeneruotų prognozių tikslumo įvertinimas (keičiant filtrų paslėptuose sluoksniuose skaičių). Po to dalyje pristatomi geriausi visų analizuojamų architektūrų modeliai, atrinkti pagal skirtingus prognozių tikslumo matavimo būdus.

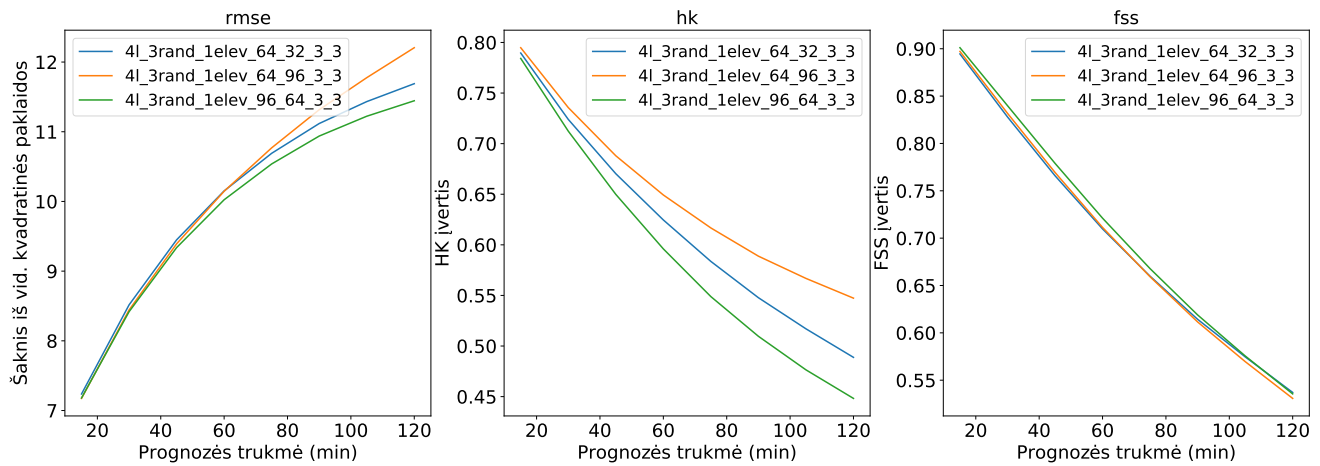
Trijų sluoksnių modeliai jau buvo analizuojami ankstesniuose eksperimentuose, tad rezultatų analizė pradedama nuo 4-ių sluoksnių CNN.

### 5.6.3 4-ių sluoksnių CNN architektūros rezultatai

Analizuojami šie parametrų deriniai:

- **I-asis modelis:** 64 filtrai pirmajame paslėptajame sluoksnyje ir 32 filtrai antrajame;
- **II-asis modelis:** 64 filtrai pirmajame ir 96 filtrai antrajame;
- **III-iasis modelis:** 96 filtrai pirmajame ir 64 filtrai antrajame.

37 pav. pateikiamas 4-ių sluoksnių CNN sugeneruotų prognozių tikslumo palyginimas pagal paslėptuosiuose sluoksniuose naudojamą filtrų kiekį.



**37 pav.** 4-ių sluoksnių CNN prognozių tikslumo pagal įvairias matavimo metrikas palyginimas

Iš palyginimo diagramų matome, kad geriausią (mažiausią) **RMSE** įvertį pasiekė **III** modelis. **HK** metrikos rezultatai visiškai priešingi: geriausias modelis pagal šį įvertį buvo **II** modelis. Pagal **FSS** įvertį visi modeliai pasirodė labai panašiai, nežymų pranašumą turėjo **III** modelis.

Neatitikimus tarp **RMSE** ir **HK** įverčių rezultatų galima paaiškinti skirtinga matavimo būdų paskirtimi: **HK** įvertis matuoja tik lietaus padengimą teritorijoje ir atsižvelgia į lietaus dažnį, o **RMSE** matavimo būdas skiria ir kritulių stiprumo neatitikimus, tačiau neįvertina teisingai teigiamų ir teisingai neigiamų prognozių dažnių skirtumo svarbos.

Nėra būdo vienareikšmiškai nustatyti, kuri iš šių matavimo metrikų yra geresnė: abi turi savo privalumų bei trūkumų ir geriausia metrika turėtų būti parenkama pagal konkrečių prognozių vartotojų poreikius.

#### 5.6.4 Autoenkoderio rezultatai

Analizuojami šie modeliai:

- **I-asis:** 32 filtrai → sutelkimas → 64 filtrai → sutelkimas → 128 filtrai → rezoliucijos padidinimas → 64 filtrai → rezoliucijos padidinimas → 16 filtrų;
- **II-asis:** 32 filtrai → sutelkimas → 64 filtrai → rezoliucijos padidinimas → 16 filtrų;
- **III-asis:** 64 filtrai → sutelkimas → 96 filtrai → sutelkimas → 192 filtrai → rezoliucijos padidinimas → 64 filtrai → rezoliucijos padidinimas → 16 filtrų.

38 pav. pateikiamas šių modelių sugeneruotų prognozių tikslumo palyginimas:

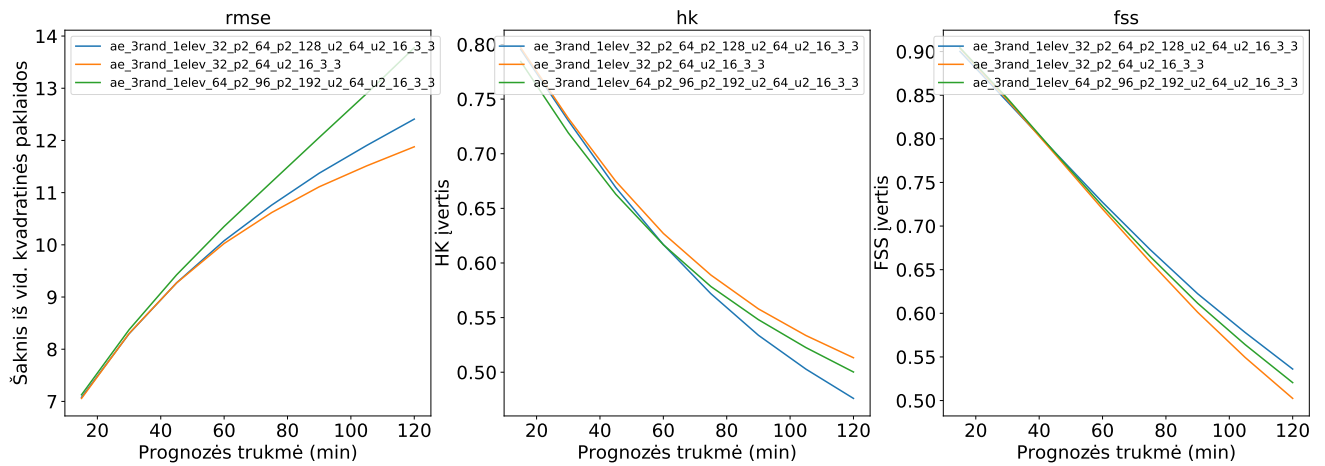
Autoenkoderio generuotų prognozių rezultatai kiek kitokie nei 4-ių sluoksnių CNN: geriausią **RMSE** įvertį turintis modelis (**II**) turėjo ir aukščiausią **HK** įvertį, tiesa nuo kitų modelių šiek tiek atsiliko pagal **FSS** metriką.

Vertinant rezultatų visumą panašu, kad gilesnis autoenkoderio tinklas (**I** ir **III** modeliai) turėjo neigiamą įtaką prognozių tikslumo rezultatams ir geram rezultatui pasiekti užteko paprastesnio modelio su mažesniu parametru kiekiu.

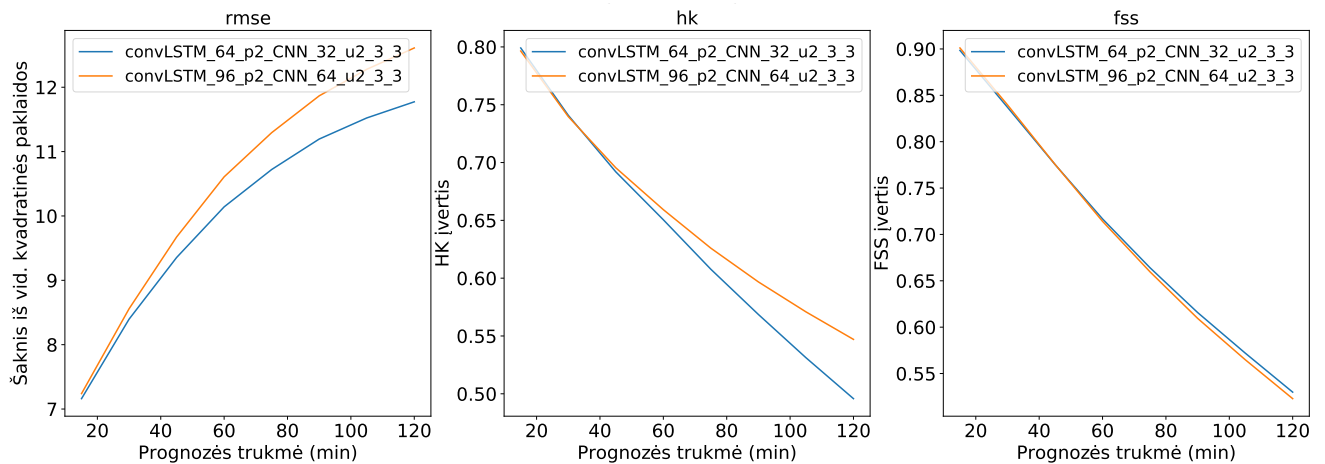
#### 5.6.5 Konvoliucinio LSTM tinklo rezultatai

Analizuojami šie modeliai:





**38 pav.** Autoenkoderio prognozių tikslumo įverčių pagal įvairias matavimo metrikas palyginimas



**39 pav.** Konvoliucinio LSTM tinklo prognozių tikslumo įverčių pagal įvairias matavimo metrikas palyginimas

- **I-asis:** convLSTM sluoksnis, 64 filtrai → sutelkimas → paprastas CNN, 32 filtrai → rezoliucijos padidinimas;
- **II-asis:** convLSTM sluoksnis, 96 filtrai → sutelkimas → paprastas CNN, 64 filtrai → rezoliucijos padidinimas;

Modelių tikslumo tarpusavio palyginimo rezultatai pateikiami 39 pav.

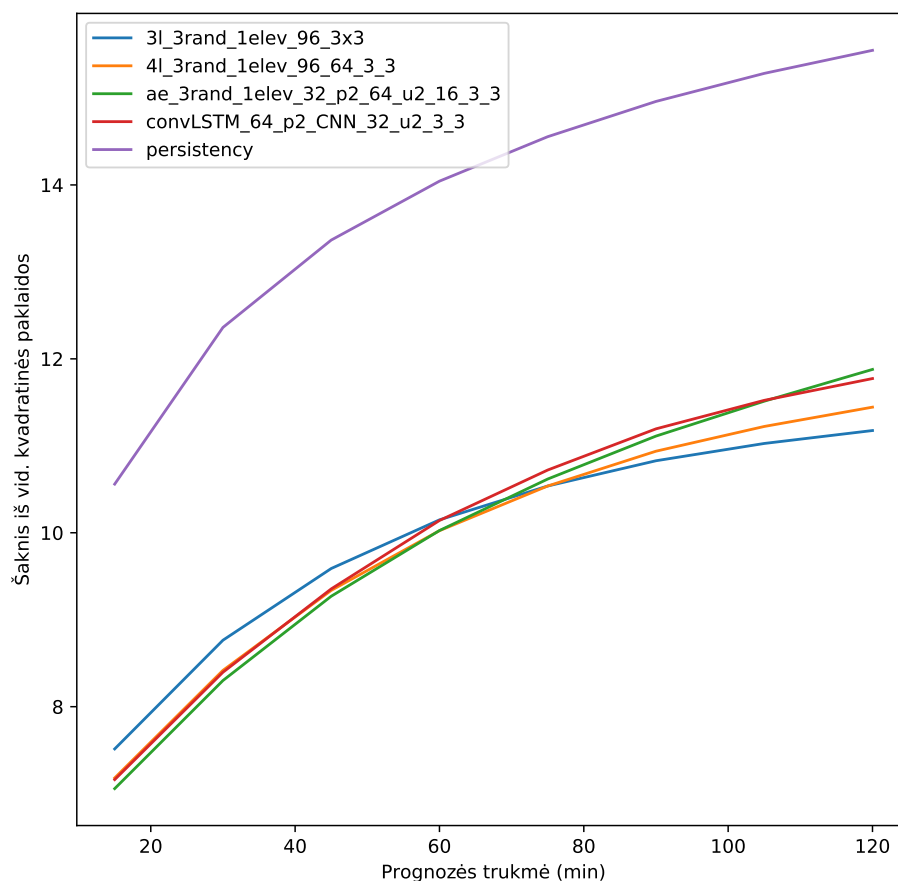
Vertinant RMSE, geriausiai pasirodė I modelis, HK – II modelis, FSS rezultatai buvo labai panašūs. Kartojasi panaši tendencija kaip ir su 4-ių sluoksnių CNN tikslumo rezultatais: pagal RMSE ir HK gaunami skirtingi geriausi modeliai.

### 5.6.6 Geriausi modeliai pagal RMSE

Metas palyginti skirtingų architektūrų modelius tarpusavyje. Šiame ir kituose skirsniuose bus pristatomi geriausi kiekvienos architektūros modeliai, parinkti pagal skirtingas prognozių tikslumo vertinimo metrikas, pradedant nuo RMSE.

40 pav. pateikiamas geriausių kiekvienos architektūros modelių palyginimas pagal RMSE metriką.

Modelių klaidos (RMSE)



40 pav. Geriausi kiekvienos architektūros modeliai pagal RMSE metriką

Iš diagramos matome, kad visų modelių tikslumas smarkiai lenkia sąlygų išsilaukymo mato įvertį. Iš analizuojamų modelių mažiausią klaidos vidurkį turėjo 4-ių sluoksnių CNN (**9,89**), nuo jo nedaug atsiliko 3-jų sluoksnių CNN (**9,95**) ir autoenkoderis (**9,97**).

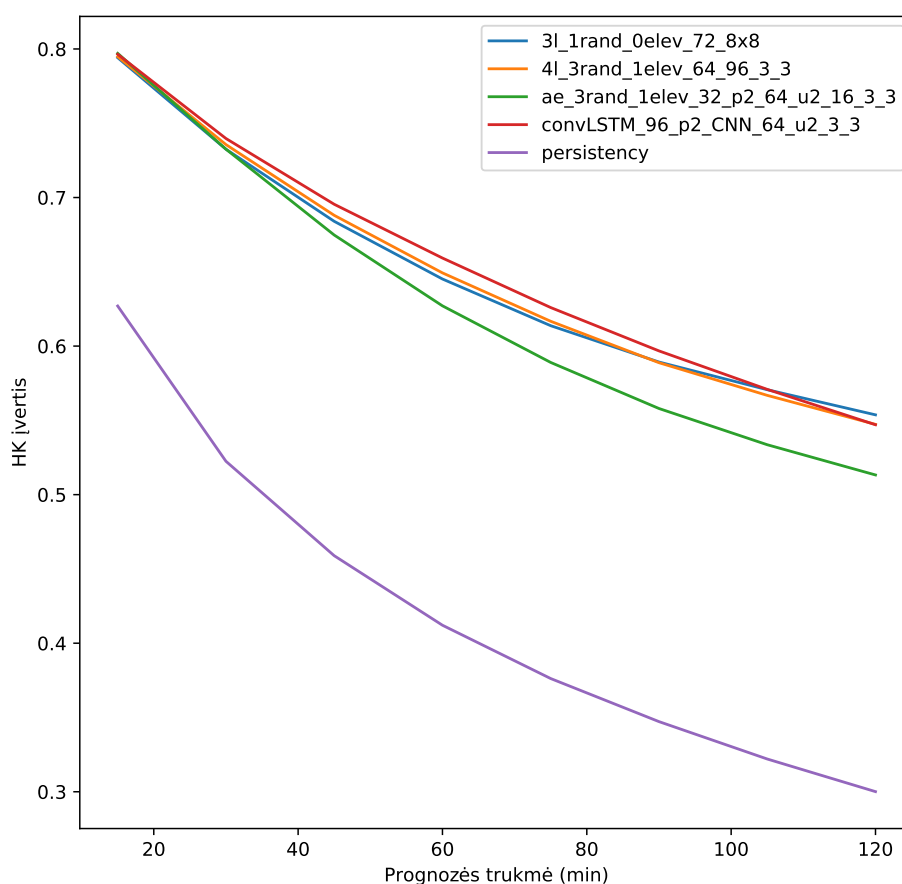
Įdomu tai, kad 3-jų sluoksnių CNN architektūros algoritmas savo konkurentams nusileido mažos trukmės prognozėse, tačiau juos aplenkė generuojant ilgesnės trukmės prognozes ( $\geq 90$  min). Šis atvejis iliustruoja eksperimento metu dažnai pastebėtą tendenciją: jei modelis duoda gerus rezultatus itin mažos trukmės prognozėms, jo tikslumas mažės bandant nuspėti ilgesnės trukmės oro sąlygas; ir atvirkščiai: modeliui demonstruojant prastesnius rezultatus 15-os ar 30-ies minučių prognozėse, jo tikslumas 1,5 ar 2 valandų prognozėms yra santykinai geresnis.

Mažiau parametų turintys modeliai savo tikslumu lenkia sudėtingesnius. Taip gali būti dėl to, kad sprendžiamai problemai dideli ir sudėtingi modeliai nereikalingi. Negalima atmesti ir galimybių, kad pasirinkti hiperparametrai ir kiti architektūriniai sprendimai neleidžia sudėtingesniems modeliams atskleisti pilno savo potencialo. Kita vertus, tai tik vienos prognozių tikslumo matavimo metodikos rezultatai.

### 5.6.7 Geriausi modeliai pagal HK įvertį

41 pav. pateikiamas geriausių kiekvienos architektūros modelių palyginimas pagal HK metriką.

## Modelių tikslumas (HK)



41 pav. Geriausi kiekvienos architektūros modeliai pagal HK metriką

HK įverčio rezultatai rodo šiek tiek kitokį vaizdą: geriausią rezultatą pasiekė konvoliucinis LSTM tinklas (HK įverčio vidurkis – **0,653**), nuo jo nežymiai atsiliko 4-ių ir 3-jų sluoksnių CNN (atitinkamai – **0,6484** ir **0,6479**). Neskaitant kiek atsilikusio autoenkoderio rezultato (**0,628**), visi modeliai parodė labai panašų tikslumą.

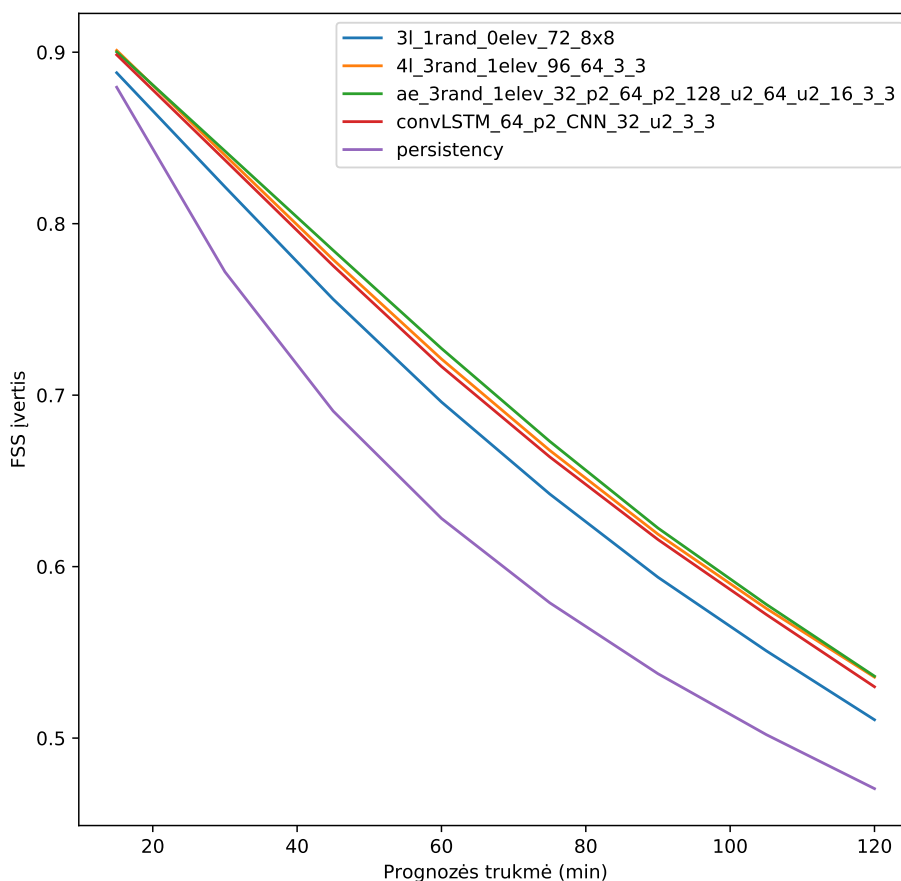
### 5.6.8 Geriausi modeliai pagal FSS įvertį

42 pav. pateikiamas geriausių kiekvienos architektūros modelių palyginimas pagal FSS metriką.

Pagal šią metriką lyderis yra autoenkoderio neuroninis tinklas (FSS įverčio vidurkis – **0,7079**), tačiau ir kitos architektūros parodė neblogus rezultatus (4-ių sluoksnių CNN – **0,7048**, 3-jų sluoksnių CNN – **0,7011**).

FSS įverčio diagrama nuo kitų išsiskiria tuo, kad modelių tikslumas laikui bėgant krenta tiesiškai, o ne kvadratinės šaknies ar eksponentinės kreivės forma. Akivaizdžių priežasčių tam surasti sunku: reikėtų gilintis į FSS įverčio skaičiavimo matematinės ypatybes.

## Modelių tikslumas (FSS)



42 pav. Geriausi kiekvienos architektūros modeliai pagal FSS metriką

## 5.7 Eksperimentų rezultatų įvertinimas

Dauguma analizuojamų neuroninių tinklų architektūrų demonstruoja panašius kritulių prognozių tikslumo rezultatus. Nors visų analizuotų architektūrų tinklai yra geriausi pagal kurią nors iš metrikų (4-ių sluoksnių CNN – RMSE, konvoliucinis LSTM – HK, autoenkoderis – FSS), tačiau visose metrikose tarp lyderių buvo 4-ių sluoksnių CNN su įvairiomis parametru konfigūracijomis.

Įvertinus apmokymui ir modelio vykdymui skirtą laiką (žr. 5-ą lentelę) ir nepriklausomai nuo pasirinkto tikslumo matavimo metodo, geriausius rezultatus pasiekė 4-ių sluoksnių CNN su 96-iais filtrais pirmajame paslėptajame sluoksnyje, 64-iais filtrais antrajame paslėptajame sluoksnyje ir  $3 \times 3$  filtro branduolio dydžiu.

Vertinant prognozių trukmę, dauguma analizuotų modelių nebuvo nuoseklūs savo tikslumu: jei rezultatai būdavo geresni generuojant mažos trukmės prognozes, tikslumas krisdavo ilgalaikiams oro sąlygų spėjimams, ir atvirkščiai. Norint išspręsti šią problemą ir dar labiau pagerinti mašininio mokymosi algoritmu tikslumą, reikėtų ansambliuoti (apjungti) kelis geriausius modelis.

Skirtingi, o kartais net priešingi, skirtingų prognozių tikslumo matavimo metodų rezultatai

rodo įvairiapusio analizuojamų modelių ištyrimo svarbą. Pasirinkus tik vieną iš visų galimų matavimo būdų galima nepastebėti svarbių tendencijų ar per anksti nurašyti naujas, iš pirmo karto nepasiteisinusias, idėjas. Pasirinktas matavimo metodas turi atitikti ir galutinių sistemos vartotojų poreikius: vienais atvejais daug svarbiau žinoti kritulių stiprumo prognozės tikslumą (stichinių reiškinių prognozavimas, žemdirbystė), o kitais – pats lietaus buvimo ar nebuvimo faktas (turizmas, sporto ir viešieji renginiai).

Visų šiame skirsnyje analizuotų algoritmų sugeneruotų prognozių pavyzdžius galima rasti 3-iajame priede. Detalūs prognozių tikslumo įverčiai pateikiami 2-ajame priede.

## 5.8 Projekto tęstinumas

Nepaisant skirtingų išanalizuotų neuroninių tinklų architektūrų kiekio, turima duomenų apimtis ir įgyta patirtis skatina atlikti dar daugiau tyrimų ir atsakyti į klausimus, kurie nebuvo paliesti šiame magistro projekte.

- Apmokant neuroninius tinklus, geriausius parametrų rinkinius pasirinkti ne tik pagal FSS įvertį, bet ir pagal kitas metrikas.
- Pasinaudoti FSS metrikos galimybėmis įvertinti skirtingų dydžių kaimynystės matmenis ir gauti geriau interpretuojamą prognozės tikslumo įvertinimą.
- Sukurtus algoritmus pritaikyti kitų teritorijų (ne tik Lietuvos) krituliams prognozuoti ir pasinaudoti daug tvarkingesniais, nuoseklesniais ir švaresniais kitų šalių kritulių radarų duomenimis, nes ne visi duomenys buvo idealūs dirbtinių neuroninių tinklų apmokymui.
- Įvestyje pridėti papildomus dinامينius parametrus, kurie modeliui galėtų padėti nustatyti kritulių susidarymo ir judėjimo tendencijas: oro temperatūrą, vėjo greitį, slėgį ir kitus įverčius.
- Rezultatus pristatyti kvalifikuotiems Lietuvos hidrometeorologijos tarnybos specialistams ir surinkti jų įžvalgas.

## 6 Išvados

Magistro projektą apibendrina šios išvados:

1. Trumpalaikės orų prognozės ir joms generuoti naudojami metodai nėra tradicinių prognozių metodų pakaitalas. Naudojantis įvairiais pradinį sąlygų ekstrapoliavimo algoritmais (įskaitant ir mašininį mokymąsi), prognozes generuoti galima žymiai greičiau, sueikvojant daug mažiau skaičiavimo resursų ir taip taupant Žemės energijos išteklius. Tiesa, tokiu būdu sugeneruotų prognozių tikslumas labai sparčiai krenta ir jų rezultatų negalima naudoti nei vidutinės, nei ilgos trukmės meteorologinių sąlygų spėjimams. Kita vertus, sparčiai tobulėjant įvairiems mašininio mokymosi algoritams ir augant skaičiavimo resursams, tikslių prognozių generavimo trukmė neišvengiamai didės.
2. Analizės dalyje bendrais bruožais susipažinta su meteorologijos mokslu, esminiais kritulių judėjimą veikiančiais fizikiniais reiškiniais bei Doplerio kritulių radaro technologija ir jos apribojimais. Net paviršutinis dalykinės srities pažinimas magistrinio projekto metu padėjo sukurti geresnius neuroninių tinklų modelius, daugumos tirtų modelių įvestyje pridant vietovės reljefo informaciją. Be to, pagrindinio duomenų šaltinio trūkumų pažinimas leido kokybiškiau įvertinti gautus rezultatus: modelių tikslumą galėjo lemti klaidingi (tušti) žemėlapių kadrai, radarų padengimo srities nepastovumas ir nepageidaujami artefaktai. Darbus tęsiant toliau, papildoma pagalba iš meteorologijos srities profesionalų būtų neįkainojama: viena svarbiausių kokybiško mašininio mokymosi modelio sėkmės paslapčių yra puikus dalykinės srities išmanymas, leidžiantis kurti dar geresnius mašininio mokymosi algoritmų modelius.
3. Dirbtinių neuroninių tinklų pagalba sugeneruotos trumpalaikės orų prognozės ne tik plačiai nagrinėjamos literatūroje, bet jau yra naudojamos ir praktikoje, prognozuojant potvynius ar staigių konvekcinių audrų susidarymą ir judėjimą. Kritulių prognozavimas iš Doplerio radarų paveikslėlių yra sritis, itin dominanti neuroninių tinklų specialistus, nes laike ir erdvėje pasiskirsčiusių duomenų analizė reikalauja inovatyvių architektūrinių sprendimų. Kartu su vaizdo įrašų kadru prognozavimu, tai yra viena iš nedaugelio sričių, kurioje plačiai taikomas konvoliucinis ilgos trumpalaikės atminties tinklas (angl. *convolutional LSTM*), tarpusavyje derinantis erdvinis (angl. *spatial*) ir laikinius (angl. *temporal*) duomenis.
4. Geriausių eksperimento metu sukurtų algoritmų prognozės atvaizduojamos kritulių prognozių peržiūros tinklapyje [kadalis.lt](http://kadalis.lt). Svetainėje pagal faktines sąlygas realiu laiku generuojamos prognozės ir kiekvienas svetainės lankytojas gali praktiškai įvertinti, kiek tikslūs yra magistro projekto metu analizuoti mašininio mokymosi algoritmai. Su statistika ar mokslu rečiau susiduriančiai visuomenės daliai toks interaktyvus kritulių prognozių puslapis ir palyginimas su faktinėmis sąlygomis leidžia daug geriau suprasti mašininio mokymosi metodų tikslumą, nei statistiniai RMSE, HK ar FSS įverčiai.
5. Per daugiau nei du metus surinktų Doplerio kritulių radaro žemėlapių analizė leido ne tik priimti efektyvesnius mašininio mokymosi algoritmų projektavimo ir rezultatų vertinimo sprendimus, tačiau pateikė ir įdomių įžvalgų apie kritulių judėjimo krypties, greičio,

dažnio ir padengimo dydžio ypatumus. Prognozių matavimo metodų analizės metu buvo susipažinta su skirtingais statistiniais ir erdviniais prognozių tikslumui matuoti skirtais metodais, kurie leido į gaunamus rezultatus pažvelgti iš įvairių kampų. Eksperimento dalyje kiekvienas matavimo metodas gavo skirtingus tiksliausius algoritmus. Tokia baigtis išryškina prognozių tikslumo matavimo būdų svarbą: tyrimui pasirinkus tik vieną iš šių metodų, galėjo būti gauti visai kitokie rezultatai.

6. Visų modelių tikslumas smarkiai viršijo bazinį sąlygų išsilaikymo matą (spėjimo, kad sąlygos nesikeis, įvertį), geriausius rezultatus pademonstravo nesudėtinga 4-ių sluoksnių CNN architektūra. Jos pranašumas dar labiau išryškėja atsižvelgus į žymiai trumpesnių modelių apmokymo laiką palyginus su sudėtingesniais algoritmais. Norint pasiekti dar didesnę mašininio mokymosi algoritmų tikslumą, verta išanalizuoti galimybę apjungti atskirus modelius į vieną: taip galėtų būti išsprendžiama eksperimentų metu išryškėjusi tikslumo netolygumo problema, kai didėjant trumpalaikių prognozių tikslumui, mažėja ilgesnio laikotarpio prognozių tikslumas. Rezultatai parodė ir papildomų statinių duomenų kanalų konvoliucinio neuroninio tinklo įvestyje naudą: papildomi atsitiktiniai ir tiriamos vietovės reljefą atitinkantys duomenys pagerino analizuotų 3-jų sluoksnių konvoliucinių neuroninių tinklų pasiekiamą tikslumą, galimai dėl sprendžiamos svorių dalinimosi problemos CNN architektūrose, kai tinklų sluoksniai negali nustatyti pikselių pozicijos paveikslėlyje.
7. Net ir atlikus detalią statistinę algoritmų tikslumo analizę, sunku pasakyti, koks yra praktinis gautų metodų tikslumas. Prognozes galima ilgai lyginti tarpusavyje ir matuoti jų statistinį atitikimą faktiniams orų sąlygų stebėjimams, tačiau norint atsakyti, kas yra gera prognozė ir kokios tikslumo įverčių reikšmės yra priimtinos, reikėtų atskiro dar didesnės apimties mokslinio darbo.
8. Šis magistro projektas – pirmasis tyrimas, analizuojantis mašininio mokymosi metodais generuojamas kritulių prognozes Lietuvos teritorijoje. Dėl ribotos tikslų prognozių generavimo trukmės, šiame darbe sukurti algoritmai dar negali pakeisti tradicinių fizikinėmis fluidų dinamikos (angl. *fluid dynamics*) dinamikos lygtimis besiremiančių orų prognozių modelių. Nepaisant to, gauti rezultatai rodo, kad patenkinamo tikslumo trumpalaikes kritulių prognozes galima generuoti beveik realiu laiku, be brangių superkompiuterių ar didelių elektros sąnaudų.

## Literatūra

- [1] J. Eggers, J. R. Lister, and H. A. Stone, “Coalescence of liquid drops,” *Journal of Fluid Mechanics*, vol. 401, pp. 293–310, 1999.
- [2] R. G. Barry and R. J. Chorley, *Atmosphere, weather and climate*. Routledge, 2009.
- [3] R. A. Houze Jr, “Stratiform precipitation in regions of convection: A meteorological paradox?” *Bulletin of the American Meteorological Society*, vol. 78, no. 10, pp. 2179–2196, 1997.
- [4] S. Strolytė and E. Rimkus, “Vėjo krypties ir greičio kaita lietuvoje 1971-2010 metais.” *Geografija*, vol. 48, no. 2, 2012.
- [5] R. Doviak and D. Zrnić, *Doppler Radar and Weather Observations*, ser. Dover Books on Engineering Series. Dover Publications, 2006. [Online]. Available: <https://books.google.lt/books?id=ispLkPX9n2UC>
- [6] J. M. Headrick and M. I. Skolnik, “Over-the-horizon radar in the hf band,” *Proceedings of the IEEE*, vol. 62, no. 6, pp. 664–673, 1974.
- [7] R. Vogt, T. Crum, J. Maj, Sandifer, E. Ciardi, and R. Guenther, “A way forward; wind farm–weather radar coexistence,” *Preprints, WINDPOWER*, 01 2009.
- [8] S. A. Gauthreaux Jr, “Weather radar quantification of bird migration,” *BioScience*, vol. 20, no. 1, pp. 17–19, 1970.
- [9] J. W. Wilson, N. A. Crook, C. K. Mueller, J. Sun, and M. Dixon, “Nowcasting thunders-torms: A status report,” *Bulletin of the American Meteorological Society*, vol. 79, no. 10, pp. 2079–2100, 1998.
- [10] P. Li and E. S. Lai, “Applications of radar-based nowcasting techniques for mesoscale weather forecasting in hong kong,” *Meteorological Applications*, vol. 11, no. 3, pp. 253–264, 2004.
- [11] L. Li, W. Schmid, and J. Joss, “Nowcasting of motion and growth of precipitation with radar over a complex orography,” *Journal of applied meteorology*, vol. 34, no. 6, pp. 1286–1300, 1995.
- [12] K. Browning and C. Collier, “Nowcasting of precipitation systems,” *Reviews of Geophysics*, vol. 27, no. 3, pp. 345–370, 1989.
- [13] J. D. Kelleher, B. Mac Namee, and A. D’arcy, *Fundamentals of machine learning for predictive data analytics: algorithms, worked examples, and case studies*. MIT press, 2015.
- [14] J. Kirchner, A. Heberle, and W. Löwe, “Classification vs. regression-machine learning approaches for service recommendation based on measured consumer experiences,” in *2015 IEEE World Congress on Services*. IEEE, 2015, pp. 278–285.



- [15] J. L. G. Rosa, “Biologically plausible artificial neural networks,” in *Artificial Neural Networks-Architectures and Applications*. IntechOpen, 2013.
- [16] J. Patterson and A. Gibson, *Deep learning: A practitioner’s approach*. ” O’Reilly Media, Inc.”, 2017.
- [17] M. Holmstrom, D. Liu, and C. Vo, “Machine learning applied to weather forecasting,” *Meteorol. Appl*, 2016.
- [18] M. Campolo, P. Andreussi, and A. Soldati, “River flood forecasting with a neural network model,” *Water resources research*, vol. 35, no. 4, pp. 1191–1197, 1999.
- [19] T. Hall, H. E. Brooks, and C. A. Doswell III, “Precipitation forecasting using a neural network,” *Weather and forecasting*, vol. 14, no. 3, pp. 338–345, 1999.
- [20] E. Hernández, V. Sanchez-Anguix, V. Julian, J. Palanca, and N. Duque, “Rainfall prediction: A deep learning approach,” in *International Conference on Hybrid Artificial Intelligence Systems*. Springer, 2016, pp. 151–162.
- [21] Y.-M. Chiang, F.-J. Chang, B. J.-D. Jou, and P.-F. Lin, “Dynamic ann for precipitation estimation and forecasting from radar observations,” *Journal of Hydrology*, vol. 334, no. 1-2, pp. 250–261, 2007.
- [22] T. Denoeux and P. Rizand, “Analysis of radar images for rainfall forecasting using neural networks,” *Neural Computing & Applications*, vol. 3, no. 1, pp. 50–61, 1995.
- [23] V. Vukotić, S.-L. Pinteá, C. Raymond, G. Gravier, and J. C. Van Gemert, “One-step time-dependent future video frame prediction with a convolutional encoder-decoder neural network,” in *International Conference on Image Analysis and Processing*. Springer, 2017, pp. 140–151.
- [24] W. Lotter, G. Kreiman, and D. Cox, “Deep predictive coding networks for video prediction and unsupervised learning,” *arXiv preprint arXiv:1605.08104*, 2016.
- [25] S. Idreos, O. Papaemmanouil, and S. Chaudhuri, “Overview of data exploration techniques,” in *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, 2015, pp. 277–281.
- [26] D. Komura and S. Ishikawa, “Machine learning methods for histopathological image analysis,” *Computational and structural biotechnology journal*, vol. 16, pp. 34–42, 2018.
- [27] M. Cappucci, “Study says ‘specific’ weather forecasts can’t be made more than 10 days in advance,” *The Washington Post*, Nov 2019. [Online]. Available: <https://www.washingtonpost.com/weather/2019/11/07/science-says-specific-weather-forecasts-cant-be-made-more-than-ten-days-advance/>
- [28] H. Das, K. Pradhan, B. R. Behera, and R. Dev, “Forecast verification analysis of rainfall and temperature for malkangiri district of odisha,” *IJCS*, vol. 6, no. 5, pp. 1731–1734, 2018.

- [29] Y. Tao, Q. Duan, A. ye, W. Gong, Z. di, M. Xiao, and K. Hsu, “An evaluation of post-processed tiggge multimodel ensemble precipitation forecast in the huai river basin,” *Journal of Hydrology*, vol. 519, 04 2014.
- [30] N. M. Roberts and H. W. Lean, “Scale-selective verification of rainfall accumulations from high-resolution forecasts of convective events,” *Monthly Weather Review*, vol. 136, no. 1, pp. 78–97, 2008.
- [31] F. Woodcock, “The evaluation of yes/no forecasts for scientific and administrative purposes,” *Monthly Weather Review*, vol. 104, no. 10, pp. 1209–1214, 1976.
- [32] O. Allouche, A. Tsoar, and R. Kadmon, “Assessing the accuracy of species distribution models: prevalence, kappa and the true skill statistic (tss),” *Journal of applied ecology*, vol. 43, no. 6, pp. 1223–1232, 2006.
- [33] S. Saseendran, S. Singh, L. Rathore, and S. Das, “Characterization of weekly cumulative rainfall forecasts over meteorological subdivisions of india using a gcm,” *Weather and Forecasting*, vol. 17, no. 4, pp. 832–844, 2002.
- [34] N. Tartaglione, “Relationship between precipitation forecast errors and skill scores of dichotomous forecasts,” *Weather and forecasting*, vol. 25, no. 1, pp. 355–365, 2010.
- [35] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.
- [36] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, “Convolutional lstm network: A machine learning approach for precipitation nowcasting,” in *Advances in neural information processing systems*, 2015, pp. 802–810.
- [37] A. Uselis, M. Lukoševičius, and L. Stasytis, “Localized convolutional neural networks for geospatial wind forecasting,” *arXiv preprint arXiv:2005.05930*, 2020. [Online]. Available: <https://arxiv.org/abs/2005.05930>

## Akronimai

**ANN** (angl. *artificial neural network*) Dirbtinis neuroninis tinklas. 10

**API** (angl. *application programming interface*) Aplikacijų programavimo sąsaja. 29

**CNN** (angl. *convolutional neural network*) Konvoliucinis neuroninis tinklas. 28, 56, 57, 59, 60

**FSS** (angl. *fraction skill score*) Dalinis tikslumo įvertis. 5, 47–50, 62, 63, 65, 67, 82

**HK** *Hanssen'o-Kuiper'io* įvertis. 5, 47–49, 62–64, 81

**LHMT** Lietuvos hidrometeorologijos tarnyba. 12, 16, 29, 31, 34, 38, 39

**LSTM** (angl. *long short-term memory*) Ilgos trumpalaikės atminties neuroniniai tinklai. 23, 28

**MSE** (angl. *mean-square error*) Vidutinė kvadratinė paklaida. 48, 50

**PI** Programinė įranga. 30–32

**RMSE** (angl. *root-mean-square error*) Šaknis iš vidutinės kvadratinės paklaidos. 5, 6, 24, 46, 47, 49, 55–60, 62, 63, 80

# Glosarijus

- Aktyvavimo funkcija** Mašiniame mokymesi, matematinė funkcija, apribojanti dirbtinio neurono išvestį funkcijos reikšmių sritimi. 21
- Autoenkoderis** Neuroninio tinklo architektūra, skirta neprižiūrimo mokymosi būdu surasti optimalų duomenų užkodavimą, kuris padeda efektyviai išskaičiuoti norimus rezultatus. 25, 27, 53
- Dalinis tikslumo įvertis** (angl. *fractions skill score*) Erdvinių prognozių įvertinimo matas. 46, 47
- Difrakcija** Bangų nuokrypis nuo tiesios judėjimo trajektorijos, užlinkstant už sutiktos kliūties. 17
- Doplerio efektas** Bangos dažnio ar ilgio kitimas priklausomai nuo judėjimo krypties, kai juda bangos šaltinis arba imtuvas. 15
- Doplerio kritulių radaras** Radaras, skirtas kritulių stebėjimui, jų judėjimo greičio, krypties bei stiprumo nustatymui. 10, 15, 27, 34
- Ekstrapoliacija** Kokio nors reiškinio, pavyzdžiui judėjimo, prognozė, remiantis prielaida, kad esamos sąlygos ir tendencijos išliks tos pačios. 10, 19, 26
- Euristika** (angl. *heuristics*) Problemų sprendimo metodas, sprendimus gaunant pasinaudojant bandymų ir klaidų būdu. 12
- Filtrų branduolio dydis** (angl. *kernel size*) Matas, konvoliuciniame neuroniniame tinkle nusakantis gretimame sluoksnyje matomų neuronų matricos dydį. 45
- Fluidų dinamika** (angl. *fluid dynamics*) Fizikinė mokslo šaka, tirianti fluidų (skysčių ir dujų) tėkmę. 12, 69
- Gilusis neuroninis tinklas** (angl. *deep neural network*) Dirbtinis neuroninis tinklas su daugiau nei vienu paslėptuoju sluoksniu. 21
- Gradientinis nusileidimas** (angl. *gradient descent*) Algoritmas, skirtas surasti daugiamatės funkcijos minimumui. 21
- Hiperparametras** Neuroninio tinklo paremetras, nustatomas prieš modelio apmokymą. 10, 25, 40, 50, 55
- HK įvertis** (angl. *Hanssen-Kuiper score*) Klaidų matrica (angl. *confusion matrix*) besiremiantis prognozių tikslumo matas. 46, 47
- Ilgos trumpalaikės atminties tinklai** (angl. *long short-term memory network, LSTM*) Rekurentiniai neuroniniai tinklai su papildomais architektūriniais sprendimais, leidžiančiais geriau išlaikyti atmintį apie seniau nutikusius įvykius. 23

**Jonosfera** Viršutinė labiau jonizuota atmosferos dalis. 17

**Kanalai** (angl. *channels*) Papildomos neuroninio tinklo įvesties dimensijos reikšmės: vaizdinio tipo įvestyje dažniausiai naudojami kaip raudonos, žalios ir mėlynos spalvos kanalų reikšmės, bet gali turėti ir kitokią paskirtį. 52

**Klaidos skleidimo atgal algoritmas** (angl. *backpropagation*) Mašiniame mokymesi, neuroninio tinklo apmokymo algoritmas, skleidžiant klaidas išvestyje atgal ir pagal tai atnaujinant arčiau išvesties esančių neuronų svorius, siekiant sumažinti suminę tinklo klaidą. 21

**Klasifikacija** Įvairias savybes turinčių objektų skirstymas į klases pagal pasirinktus bendrus požymius. 20

**Konvekcija** Meteorologijoje – makroskopiniai atmosferos dujų srautai, atsirandantys dėl temperatūros, tankio ar slėgio nevienodumo. 14

**Konvoliucinis neuroninis tinklas** (angl. *convolutional neural network, CNN*) Dirbtinio neuroninio tinklo architektūra, kai kaimyninių sluoksnių neuronai mato tam tikrą neuronų teritoriją/zoną prieš tai buvusiame sluoksnyje. 22

**Lietaus Šešėlis** Meteorologijoje – kalnų grandinės sukeltas klimatologinis reiškinys, kai vienoje kalnyno pusėje, iš kurios dažniausiai ateina krituliai vyrauja drėgnas klimatas, o kitoje – sausas. 14

**Neurono polinkis** (angl. *bias*) Mašiniame mokymesi, neurono įvertis sumuojamas su svorių ir įvesties sandauga. 21

**Okliuzija** Šilto ir šalto atmosferos frontų susidūrimas. 13, 14

**Orografiniai krituliai** Krituliai, susidarantys dėl Žemės paviršiaus aukščių skirtumų. 14, 15

**Padidinimas** (angl. *padding*) Konvoliucinio neuroninio tinklo filtro branduolio dydžio išplėtimas kraštuose, siekiant pakeisti sluoksnio dimensijų dydį išvestyje. 52

**Polarizacija** Skersinių bangų savybė, apibūdinanti virpesių orientaciją plokštumoje, statmenoje sklidimo kryptčiai. 16

**Regresija** Atsitiktinio dydžio vidurkio priklausomybės nuo kitų dydžių išraiška. 13, 20

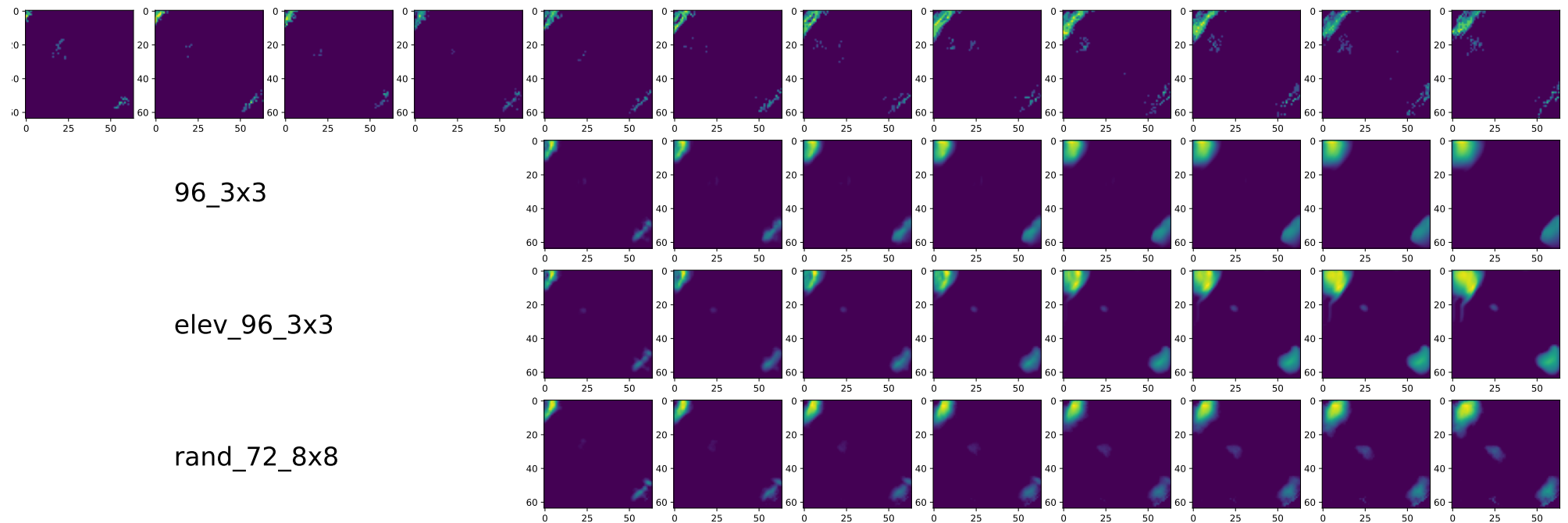
**Rekurentinis neuroninis tinklas** (angl. *recurrent neural network, RNN*) Dirbtinio neuroninio tinklo architektūra su ciklais, skirtais simuliuoti neuroninio tinklo atmintį. 22, 25

**Retoji matrica** Matrica, kurioje didžioji dalis elementų yra 0. 41

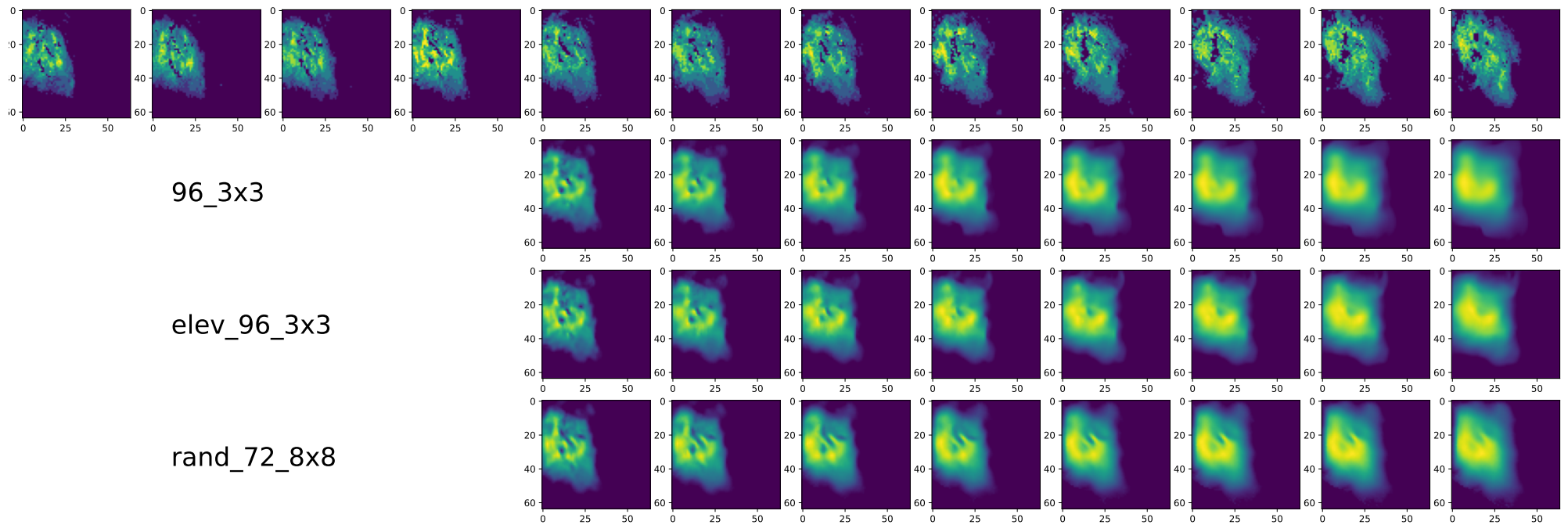
**Rumbas** Nesisteminis plokščiojo kampo matavimo vienetas, lygus 0,19635 radiano, arba  $\frac{1}{8}$  stačiojo kampo. 15, 44

- Savybių išskyrimas** (angl. *feature extraction*) Apmokymui reikalingų duomenų savybių išskyrimas mašiniame mokymesi. 20, 53
- Skatinamasis mokymasis** (angl. *reinforcement learning*) Mašininio mokymosi rūšis, kai modelis yra „apdovanojamas“ už rezultatus, artėjančius prie norimo tikslo; ir atvirkščiai – „nubaudžiamas“ už rezultatus, kurie tolsta nuo siekiamo tikslo. 20
- Sutelkimo imant maksimalią reikšmę sluoksnis** (angl. *max pooling*) Konvoliucinio neuroninio tinklo sluoksnis, atliekantis didžiausios reikšmės nustatymą prieš tai buvusio sluoksnio filtro branduolyje ir sumažinantis pločio ir aukščio dimensijas. 52
- Sąlygų išsilaikymo matas** (angl. *persistence*) Įvertis, kuris priima, kad šiuo metu esančios sąlygos ir rezultatai laikui bėgant nepakis ir išliks tokie patys. 26, 64
- Trumpalaikės orų prognozės** (angl. *nowcasting*) Tikslios, mažos trukmės orų prognozės, paprastai iki 4 valandų. 18
- Virtualizacija** (angl. *virtualisation*) Programų inžinerijoje – technologija, kuri leidžia fizinius resursus pateikti kaip loginius. 36
- Šaknis iš vidutinės kvadratinės paklaidos** (angl. *root-mean-square error, RMSE*) Statistinis įvertis, naudojamas neatitikimui tarp spėjamo/išmatuoto rezultato ir žinomo faktinio įverčio matuoti. 46

## 1 priedas. Atvejo analizė: trijų sluoksnių CNN su vienu papildomu statiniu duomenų sluoksniu

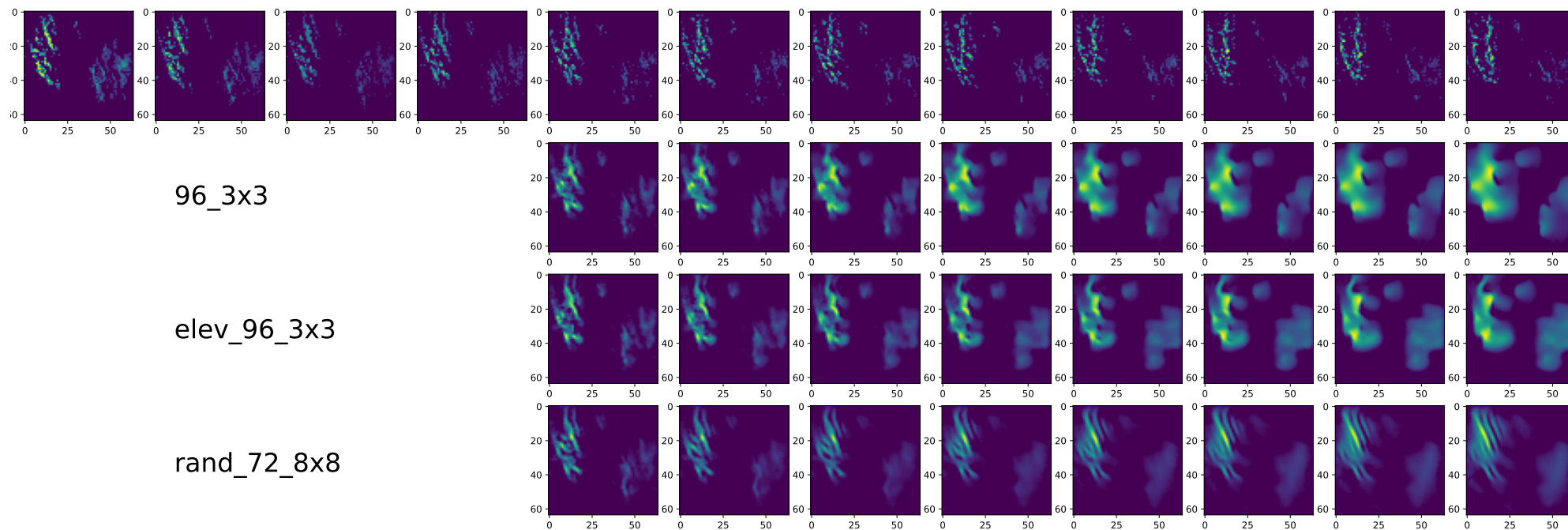


**43 pav.** Trijų sluoksnių CNN generuojamų prognozių palyginimas. Iš eilės nuo viršaus: pirmi keturi paveikslėliai viršuje – visų tinklų įvestis, t.y. žinomi faktiniai duomenys prieš prognozę. Toliau: 1 eilutė – faktiniai stebėjimo duomenys; 2 eilutė – CNN be papildomo kanalo; 3 eilutė – CNN su papildomu reljefo kanalu; 4 eilutė – CNN su papildomu atsitiktinių duomenų kanalu. Pradžioje modeliai generuoja labai panašius rezultatus, tačiau didėjant prognozės trukmei jų generuojami žemėlapiai šiek tiek išsiskiria, tačiau išlieka vizualiai artimi pradinėms sąlygoms.



**44 pav.** Trijų sluoksnių CNN generuojamų prognozių palyginimas. Visi algoritmai generuoja labai panašius rezultatus. Laikui bėgant paveikslėliai tampa vis labiau susilieję. Visi modeliai praranda kritulių sankaupos viduryje esančias skyles, tačiau išlaiko panašią sankaupos formą. Esminiai skirtumai tarp modelių – sankaupos viduryje prognozuojamų kritulių stiprumas.





**45 pav.** Trijų sluoksnių CNN generuojamų prognozių palyginimas. Šiuo atveju modeliai generuoja skirtingas prognozes ir nutolsta nuo faktinių duomenų. Realybei vizualiai artimiausia CNN su papildomu atsitiktinių duomenų kanalu modelio prognozė.

## 2 priedas. Įvairių architektūrų prognozių tikslumo įvertinimo rezultatų suvestinė

**6 lentelė** Analizuotų architektūrų RMSE įverčio rezultatai. Naudojami trumpiniai: rmseavg – įverčių vidurkis, rmse\_N – N minučių trukmės prognozės įvertis; ae - autoenkoderis; Xrand, Xelev – atitinkamo tipo papildomų kanalų skaičius; p2 - maksimalios reikšmės sutelkimo operacija; u2 - rezoliucijos padidinimo operacija. Pastaba: autoenkoderio tinklas taip pat naudoja 3 atsitiktinius ir 1 reljefo statinį kanalą, tačiau dėl vietos lentelėje trūkumo modelio pavadinimas sutrumpintas

Modelis	rmseavg	rmse_15	rmse_30	rmse_45	rmse_60	rmse_75	rmse_90	rmse_105	rmse_120
4l_3rand_1elev_96_64_3_3	9,888	7,180	8,418	9,337	10,024	10,538	10,938	11,222	11,445
3l_3rand_1elev_96_3_3	9,948	7,514	8,764	9,589	10,148	10,538	10,828	11,027	11,176
ae_32_p2_64_u2_16_3_3	9,972	7,058	8,301	9,270	10,026	10,618	11,112	11,513	11,878
3l_1rand_0elev_72_8_8	10,030	7,273	8,518	9,437	10,117	10,644	11,077	11,426	11,747
convLSTM_64_p2_CNN_32_u2_3_3	10,033	7,161	8,396	9,354	10,141	10,721	11,195	11,523	11,773
4l_3rand_1elev_64_32_3_3	10,035	7,237	8,516	9,449	10,151	10,691	11,118	11,433	11,689
ae_32_p2_64_p2_128_u2_64_u2_16_3_3	10,146	7,077	8,296	9,272	10,078	10,761	11,372	11,904	12,409
4l_3rand_1elev_64_96_3_3	10,152	7,174	8,442	9,391	10,146	10,769	11,316	11,776	12,206
convLSTM_96_p2_CNN_64_u2_3_3	10,516	7,240	8,557	9,673	10,608	11,294	11,866	12,278	12,610
ae_64_p2_96_p2_192_u2_64_u2_16_3_3	10,652	7,127	8,373	9,425	10,352	11,206	12,057	12,899	13,776
persistence	13,834	10,559	12,361	13,366	14,043	14,555	14,960	15,283	15,548

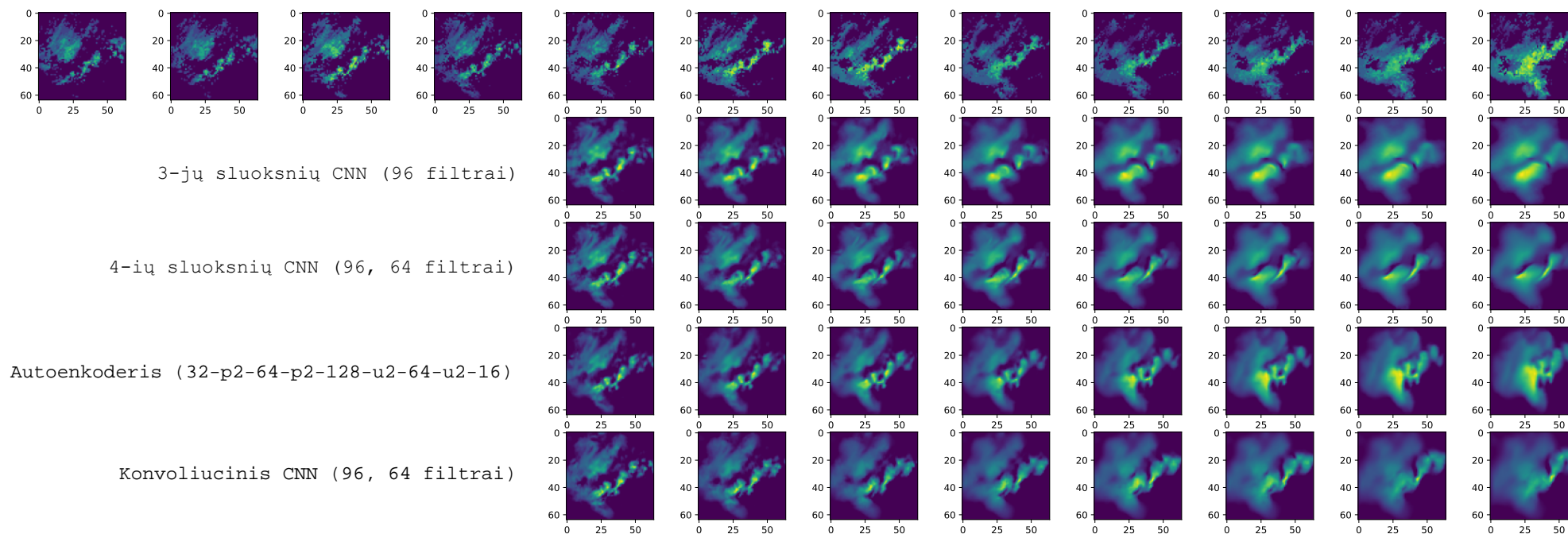
**7 lentelė** Analizuotų architektūrų HK įverčio rezultatai. Naudojami trumpiniai: hkavg – įverčių vidurkis, hk\_N – N minučių trukmės prognozės tikslumo įvertis; ae - autoenkoderis; Xrand, Xelev – atitinkamo tipo papildomų kanalų skaičius; p2 - maksimalios reikšmės sutelkimo operacija; u2 - rezoliucijos padidinimo operacija. Pastaba: autoenkoderio tinklas taip pat naudoja 3 atsitiktinius ir 1 reljefo statinį kanalą, tačiau dėl vietos lentelėje trūkumo modelio pavadinimas sutrumpintas

Modelis	hkavg	hk_15	hk_30	hk_45	hk_60	hk_75	hk_90	hk_105	hk_120
convLSTM_96_p2_CNN_64_u2_3_3	0,654	0,796	0,740	0,695	0,659	0,626	0,597	0,571	0,547
4l_3rand_1elev_64_96_3_3	0,648	0,795	0,736	0,688	0,649	0,617	0,589	0,567	0,547
3l_1rand_0elev_72_8x8	0,648	0,794	0,733	0,684	0,645	0,614	0,589	0,571	0,554
3l_3rand_1elev_96_3x3	0,639	0,783	0,718	0,669	0,633	0,605	0,583	0,566	0,552
convLSTM_64_p2_CNN_32_u2_3_3	0,636	0,799	0,741	0,692	0,651	0,608	0,569	0,531	0,496
ae_32_p2_64_u2_16_3_3	0,628	0,797	0,733	0,675	0,627	0,589	0,558	0,534	0,513
4l_3rand_1elev_64_32_3_3	0,618	0,790	0,724	0,670	0,625	0,584	0,548	0,517	0,489
ae_64_p2_96_p2_192_u2_64_u2_16_3_3	0,617	0,784	0,719	0,663	0,617	0,578	0,548	0,523	0,500
ae_32_p2_64_p2_128_u2_64_u2_16_3_3	0,612	0,795	0,730	0,669	0,617	0,572	0,534	0,503	0,476
4l_3rand_1elev_96_64_3_3	0,591	0,784	0,712	0,650	0,596	0,549	0,510	0,476	0,448
persistence	0,421	0,627	0,522	0,459	0,412	0,376	0,347	0,322	0,300

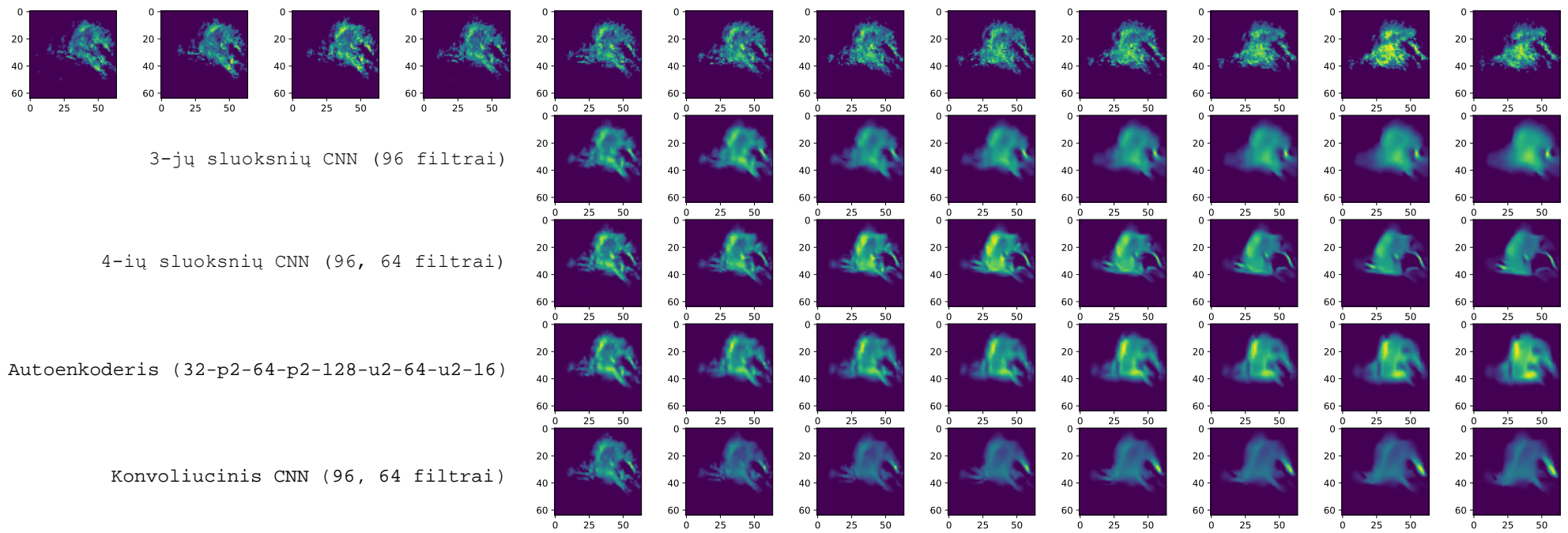
**8 lentelė** Analizuotų architektūrų FSS įverčio rezultatai. Naudojami trumpiniai: fssavg – įverčių vidurkis, fss\_N – N minučių trukmės prognozės tikslumo įvertis; ae - autoenkoderis; Xrand, Xelev – atitinkamo tipo papildomų kanalų skaičius; p2 - maksimalios reikšmės sutelkimo operacija; u2 - rezoliucijos padidinimo operacija. Pastaba: autoenkoderio tinklas taip pat naudoja 3 atsitiktinius ir 1 reljefo statinį kanalą, tačiau dėl vietos lentelėje trūkumo modelio pavadinimas sutrumpintas

Modelis	fssavg	fss_15	fss_30	fss_45	fss_60	fss_75	fss_90	fss_105	fss_120
ae_32_p2_64_p2_128_u2_64_u2_16_3_3	0,708	0,900	0,842	0,784	0,727	0,673	0,622	0,578	0,536
4l_3rand_1elev_96_64_3_3	0,705	0,901	0,840	0,779	0,721	0,668	0,619	0,576	0,536
ae_64_p2_96_p2_192_u2_64_u2_16_3_3	0,702	0,904	0,846	0,784	0,723	0,665	0,612	0,564	0,521
convLSTM_64_p2_CNN_32_u2_3_3	0,701	0,898	0,837	0,775	0,717	0,664	0,616	0,572	0,530
convLSTM_96_p2_CNN_64_u2_3_3	0,698	0,901	0,840	0,776	0,714	0,660	0,610	0,565	0,523
4l_3rand_1elev_64_32_3_3	0,698	0,895	0,828	0,766	0,710	0,660	0,614	0,575	0,537
4l_3rand_1elev_64_96_3_3	0,698	0,897	0,832	0,769	0,711	0,659	0,612	0,570	0,531
ae_32_p2_64_u2_16_3_3	0,695	0,904	0,844	0,782	0,720	0,659	0,601	0,549	0,502
3l_1rand_0elev_72_8x8	0,682	0,888	0,821	0,756	0,696	0,642	0,594	0,551	0,511
3l_3rand_1elev_96_3x3	0,679	0,882	0,810	0,746	0,687	0,637	0,593	0,554	0,519
persistence	0,632	0,880	0,772	0,691	0,628	0,579	0,538	0,502	0,471

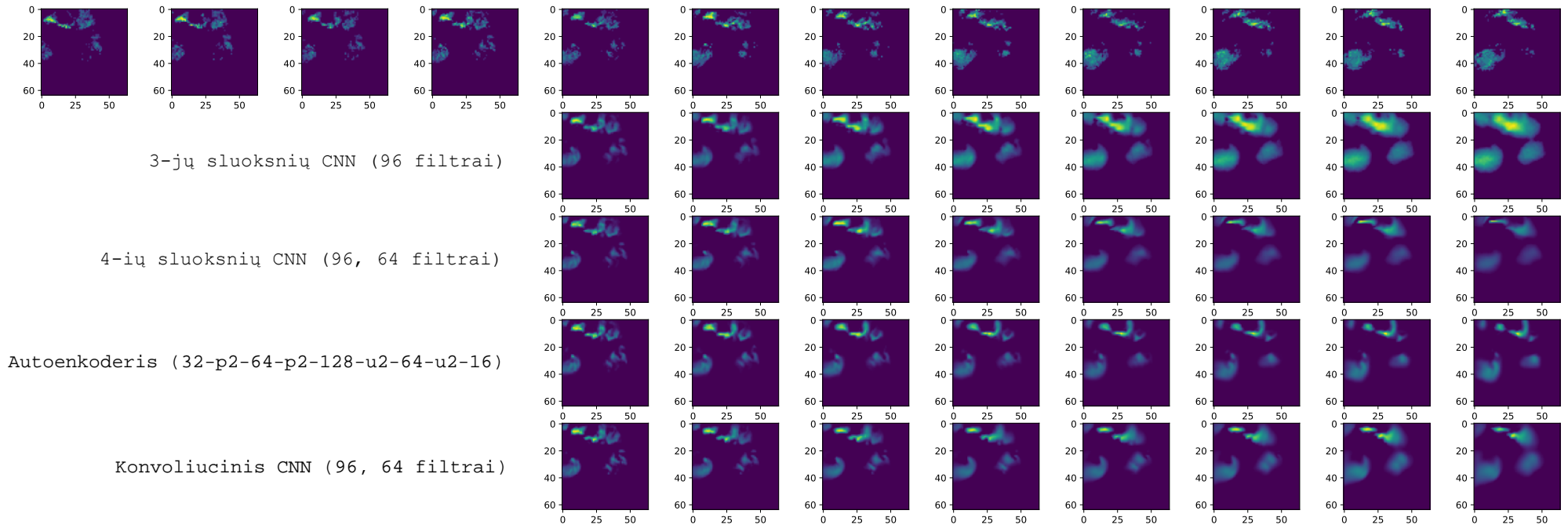
### 3 priedas. Atvejo analizė: įvairių CNN architektūrų generuojamų prognozių pavyzdžiai



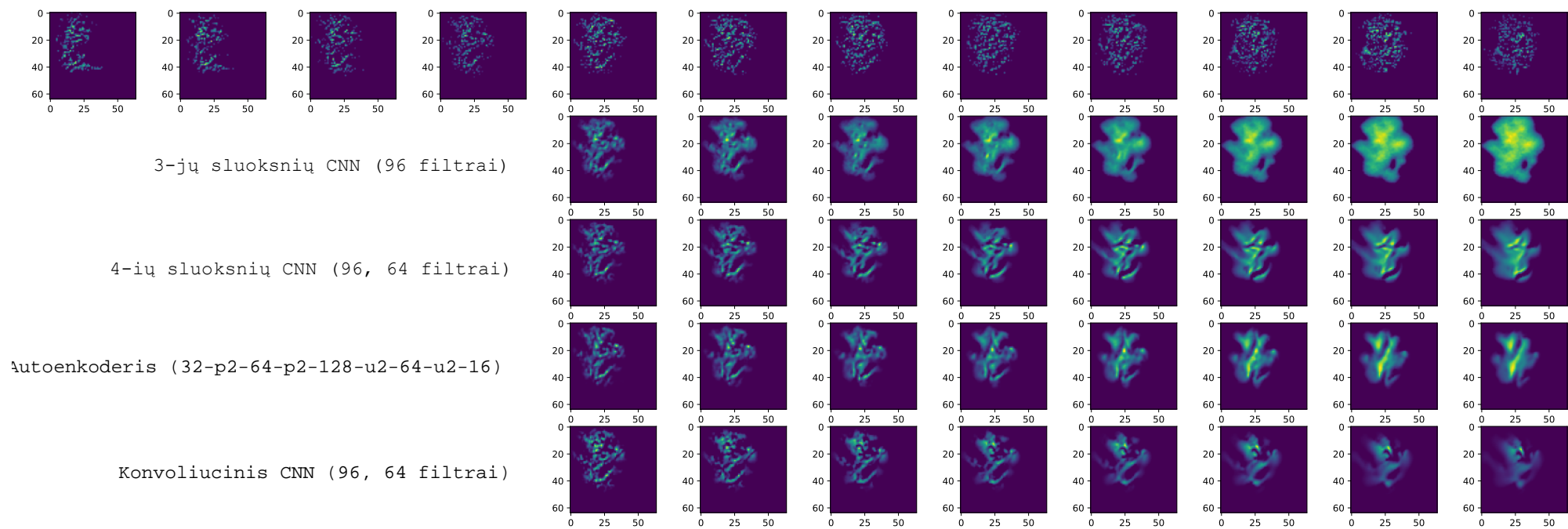
**46 pav.** Įvairių architektūrų CNN generuojamų prognozių palyginimas: 1-as atvejis, itin lietinga 2020 m. gegužės 11-a diena; pirmi keturi paveikslėliai viršuje – visų tinklų įvestis, t.y. žinomi faktiniai duomenys prieš prognozę, toliau: 1 eilutė – faktiniai stebėjimo duomenys; kitos eilutės – skirtingų algoritmų sugeneruotos prognozės



**47 pav.** Įvairių architektūrų CNN generuojamų prognozių palyginimas: 2-as atvejis, 2020 m. gegužės 05-a diena; retesnis atvejis, kai krituliai atkeliauja iš rytų pusės. Panašu, kad trijų sluoksnių CNN neatpažino kritulių judėjimo krypties

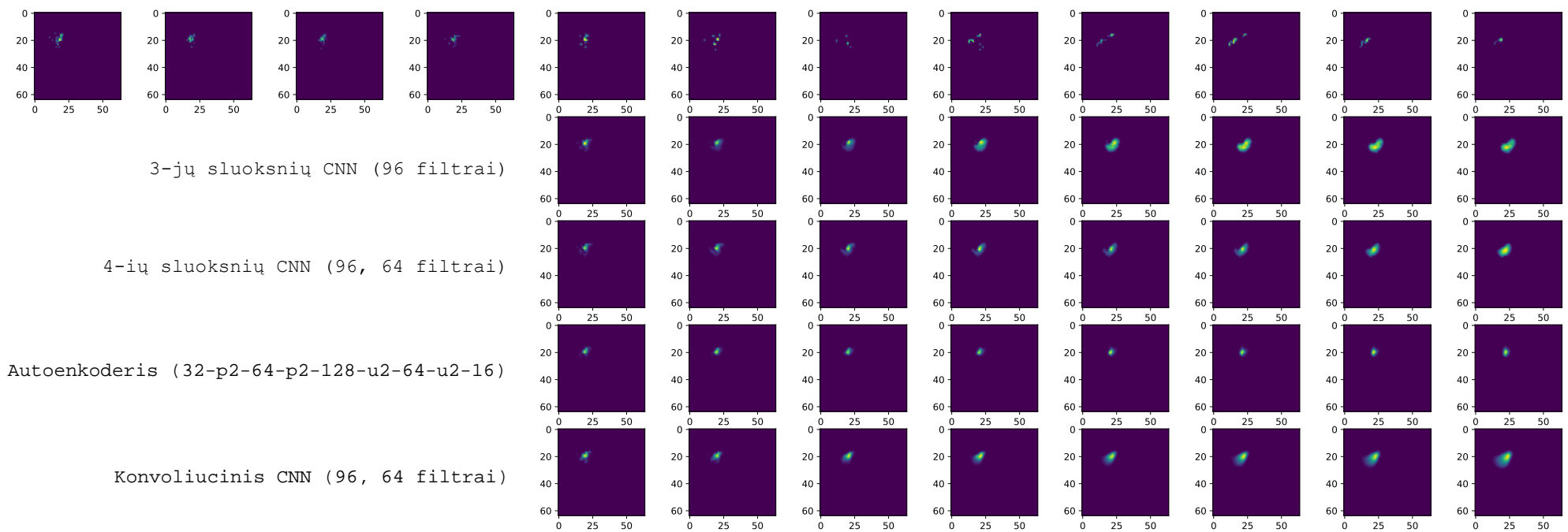


**48 pav.** Įvairių architektūrų CNN generuojamų prognozių palyginimas: 3-as atvejis, 2020 m. balandžio 28-a diena. Iš vakarų pasirodo nauja kritulių masė. Dauguma tinklų atpažįsta, kad ji iš lėto didės ir judės į rytus. Trijų sluoksnių CNN prognozavo žymiai platesnį kritulių srities padengimą nei fiksuota iš tikrųjų



**49 pav.** Įvairių architektūrų CNN generuojamų prognozių palyginimas: 4-as atvejis, 2020 m. balandžio 14-a diena. Pavieniai išsibarstę debesys. Ši atvejį sunku prognozuoti visiems modeliams: bėgant laikui prognozėse smulkūs debesų susilieja į vieną





**50 pav.** Įvairių architektūrų CNN generuojamų prognozių palyginimas: 5-as atvejis, 2020 m. sausio 19-a diena. Smulkių kritulių prognozė. Tiksliausiai kritulių srities dydį prognozuoja autoenkoderis

## **4 priedas. Pirmieji trumpalaikių kritulių prognozių tyrimo, naudojantis Doplerio radaro kritulių radaro duomenis Lietuvoje, rezultatai**

Tyrimas anglų kalba, pristatytas konferencijoje IVUS 2018 ir publikuotas žurnale: <http://ceur-ws.org/Vol-2147/p10.pdf>.

# Nowcasting precipitation using weather radar data for Lithuania: the first results

1<sup>st</sup> Aivaras Čiurlionis  
Faculty of Informatics  
Kaunas University of Technology  
Kaunas, Lithuania  
aivaras.ciurlionis@ktu.edu

2<sup>nd</sup> Mantas Lukoševičius  
Faculty of Informatics  
Kaunas University of Technology  
Kaunas, Lithuania  
mantas.lukosevicius@ktu.lt

**Abstract**— Although the accuracy and the duration of modern weather forecasts constantly increase together, numerical weather prediction methods still face a few drawbacks. Due to an extensive computing time and a high power usage, these methods are unable to efficiently react to rapidly changing initial weather conditions. Also, most of the numerical weather prediction models can be less accurate for smaller regions with specific local weather conditions. These problems are addressed by a technique called nowcasting, which uses an extrapolation of various current weather conditions. Multiple research papers have shown that this technique can outperform traditional weather predictions for up to two hours. Furthermore, it can be improved using machine learning algorithms. In this paper nowcasting algorithms are used to predict a short-term precipitation over Lithuania using weather radar images provided by Lithuanian Hydrometeorology service. A Hanssen–Kuipers score is used to evaluate the accuracy of prediction against observed precipitation maps. The results of three extrapolation algorithms (basic translation, step translation, and sequence translation) and a single machine learning algorithm based on convolutional neural networks (CNN) are evaluated for two chosen hours and compared to the persistency algorithm. The average scores of each prediction algorithm for a single week are also presented. Although the results remain accurate for up to 45 minutes only, the accuracy can be improved by adding additional variables to the extrapolation. The better accuracy can also be achieved by using more sophisticated machine learning algorithms, like recurrent neural networks and their variations, that take dependencies on previous inputs in time series into account. This paper presents the first results of the algorithms, which are to be improved by further research.

**Keywords**—*meteorology, precipitation, forecast, nowcasting*

## I. INTRODUCTION

Due to the steadily growing computational capabilities of modern computers during the recent years, the accuracy and the duration of weather forecast has increased. The accuracy of the current official Day 5–7 forecasts is found to be similar to that of Day-1 forecasts from 50 years ago [1].

However, the amount of computational resources required for the evaluation of complex weather prediction models is also constantly rising. In order to achieve a weather forecast that is accurate and up to date, weather prediction services are using some of the most powerful supercomputers in the world. These computers require a high amount of power and other resources for operation and cooling.

Moreover, the time required to collect the data from weather observation stations, to perform all the calculations and to post-process and visualize these results might take hours. Most of the weather prediction models are global and, in order to adapt these results for local conditions in their region, further processing by professionals from regional weather services from is required. This means that forecasters can fail to predict rapid changes in weather, such as sudden convective summer storms, hurricanes, and flooding, since an event may occur before forecast calculations are completed.

This issue is addressed by using a nowcasting, which is defined as the weather forecasts on very short-term period of up to 2 hours. Nowcasting is an extrapolation of current known weather conditions such as a current temperature, cloud coverage, satellite data and other parameters. A Doppler's weather radar can be used to extrapolate precipitation amplitude and location.

Nowcasting techniques are considerably faster than complex numerical weather forecast models and can be applied to predict a rapidly changing weather conditions. Nowcasting can also be used to improve existing weather forecast models by introducing more accurate data for short-term regional weather prediction and implement more precise weather alert systems that can potentially save people's lives by warning about unexpected rapidly forming storms and possible flooding.

This paper presents the first results of precipitation prediction algorithms that use weather radar images for nowcasting. The algorithms used to predict a movement of precipitation systems, use simple extrapolation and machine learning techniques, however, the obtained knowledge and results will be used to build a more complex and more accurate prediction system.

## II. RELATED WORK

In this section related work on weather data extrapolation and other short-term weather prediction methods will be discussed.

Li, Schmid, and Joss define two major extrapolation techniques: one technique tries to find the best possible fit between two different maps of radar data. The correlation coefficient is used as an objective test criterion for the agreement between the two radar patterns. The mean vector of displacement, that can be found from the observed radar pattern, allows a linear extrapolation into the future. Another group of nowcasting techniques has the ability to track and forecast the

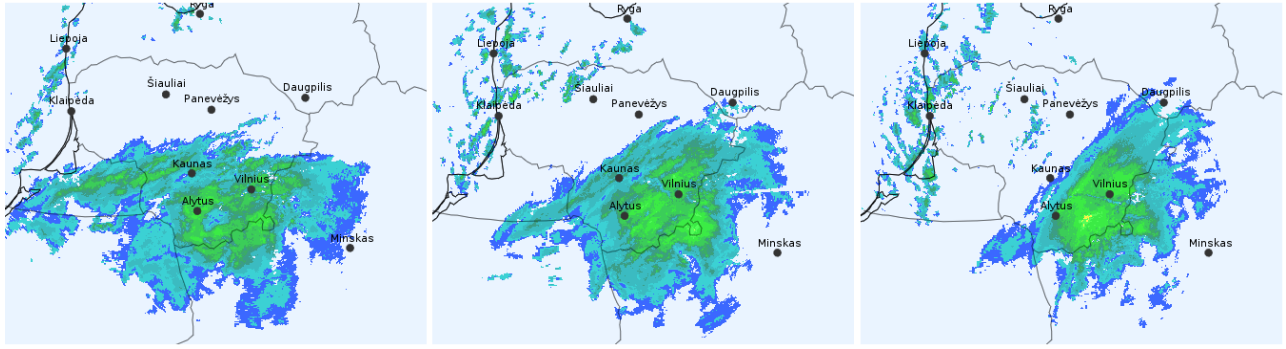


Fig. 1. The weather radar maps over Lithuania on 29/10/2017 from 11:15 to 15:15 every two hours. The blue shades indicate low rates of rain, while the green ones show a higher volume of precipitation.

areas, the mass centroids, or other parameters of closed radar contours that represent individual convective storms or cells. Detection of a motion vector can allow further extrapolation of other storm parameters [2].

For example, Saxen et. all adapted extrapolation method to forecast thunderstorm initiation, growth, and decay. This technology is used in real-world military level applications to ensure the safety of personnel that works on the missile range [3].

Worth noting is that nowcasting and extrapolation techniques are also used for public weather prediction. It is especially useful where complex local orography and a high convective activity limit global prediction models' accuracy. Li and Lai describe such a system in Hong Kong. Two methods are being used: the first one is object-oriented, where pixels in the radar images are grouped over some predefined intensity threshold in the form of an ellipse and tracks the movement of ellipse centroids between successive radar images. The other one derives vectors from the matching of pixel arrays (boxes) between two successive radar images through maximum cross-correlation [4].

According to Li and Lai, this system has enabled forecasters to make qualitative educated guesses of the likelihood of prolonged heavy rain or the potential of enhanced storm development. [4].

Wilson, Crook, Mueller, Sun, and Dixon in their paper Nowcasting Thunderstorms: A Status Report review the status of forecasting convective precipitation for time periods less than a few hours. In their review of nowcasting thunderstorm location by extrapolating radar echoes they state that the accuracy of these forecasts generally decreases very rapidly during the first 30 min because of the very short lifetime of individual convective cells. Fortunately, more organized features like squall lines and supercells can be successfully extrapolated for the longer time period [5].

Comparing persistency and extrapolation methods for 30 min. forecasts, Wilson et. all stated that probability of detection (POD) for persistency method is 0.13, and 0.27 for extrapolation, while false alarm ratio (FAR) is 0.85 and 0.59 respectively. These results show that extrapolation method can be significantly more accurate than a basic (often rather precise) persistency method.

Adding to what has been mentioned previously, some interesting applications of machine learning algorithms in the weather prediction area can be found. Holmstrom, Liu, and Vo implemented linear regression solution to forecast the lowest and the highest day temperature. However, the evaluation results have shown, that for a short forecast algorithm's mean squared error is almost twice as big as the error of professional forecasts [6].

Despite this, there was also some promising application: Campolo, Andreussi, and Soldati were highly satisfied with their results of predicting river flooding with a neural network model. [7].

Furthermore, Denoex and Rizand, have developed a machine learning solution based on a neural network model for a precipitation prediction from weather radar images. Authors state that, although more experiments in various meteorological situations are still needed to complete the validation of this approach, the results obtained so far are considered as highly encouraging. Their algorithm managed to outperform both persistency and extrapolation (cross-correlation) methods in short-term forecasts [8].

### III. THE DATA

The data for this research are taken from publicly available factual weather radar maps provided by the Lithuanian Hydrometeorology service (Fig. 1).

The maps are generated every 15 minutes and indicate the observed amount of precipitation that is captured by the Doppler's weather radar. The maps cover all area of Lithuania and display a combined result of the data from two weather radars: one in Laukuva (Western Lithuania), the other in Traku Vokė (Eastern Lithuania). Each pixel in a map represents one of 16 different levels of precipitation: a level of 0 indicates no precipitation over the area, while level 16 shows extremely high precipitation of more than 66 mm/h.

Although the precipitation data from the weather radars can be interpreted as an actual observed rainfall in a given area, there are some limitations that should be considered. First of all, the weather maps do not differentiate between the types of precipitation. Whether it is rain, snow, or hail, it will have the same representation in a map. Secondly, the further an area is from a radar, the lower resolution is available. Although such

decrease in resolution is not significant for Lithuania, it might result in a lower accuracy for the regions that are further away from the radars. Not every object detected by a radar is precipitation. For example, mountains, high buildings, wind farms [9] or even bird migration [10] can be mistaken for a rain or block a field of view to the actual precipitation.

In this research, weather radar data from the date range of 23/10/2017 to 30/10/2017 will be used for the evaluation of algorithms. This week contains three major precipitation events and periods without rain between them.

The radar images over Lithuania are available at a maximum resolution of 768 by 768 pixels, but due to performance reasons (especially for machine learning algorithms), all images are scaled down to a resolution of 64 by 64 pixels.

Convolutional neural network (CNN) based algorithm was trained with 10 000 sets of weather radar images, that were retrieved between 01/11/2017 and 25/04/2018.

#### IV. EVALUATION OF ACCURACY

Evaluating the accuracy of precipitation forecast is rather a challenge. A forecast is accurate only if predicted precipitation closely matches an actual observed rainfall amount. In this research, weather radar images are used both as prediction source and as an evaluation.

To compare the similarity between two images, one can use a traditional root mean square error algorithm, where the difference between actual and observed precipitation amount is calculated. However, the results of this error function have no clear boundaries and it is hard to evaluate how accurate a forecast from an error value actually is.

For this reason, in this paper, the Hanssen–Kuipers (HK) score, also known as the true skill statistic, is used. This score describes the performance of a classification model and is widely used for forecast verification [11].

First, each grid-point (pixel) in an actual and predicted precipitation map is classified into four categories: correct non-rain forecasts (Z), false alarms (F, precipitation in a certain area was predicted, but did not occur), misses (M, the precipitation was not predicted, but did occur), or hits (H, a precipitation event was predicted successfully).

From the number of grid points in each category, it is possible to calculate the HK score using Equation (1).

$$HK = \frac{(ZH-FM)}{(Z+F)(M+H)} \quad (1)$$

This score can fall between 0 and 1, where a score of 1 indicates an ideal forecast.

However, the HK score only uses the occurrence of a rain event without taking the strength of precipitation into account. This means that a predicted rainfall amount of 1 mm/h, while the actual was 30 mm/h, would be considered as a hit. Furthermore, only respective pixels in an actual and predicted image are compared. If a rainfall event did actually occur, but just a few pixels away, this would be considered as a miss or a false alarm.

These problems can be addressed by introducing the precipitation strength thresholds during the classification and by increasing a score for near misses. Nevertheless, this research is oriented to a comparison of different algorithms only and any adjustments of score would be unnecessary.

#### V. ALGORITHMS

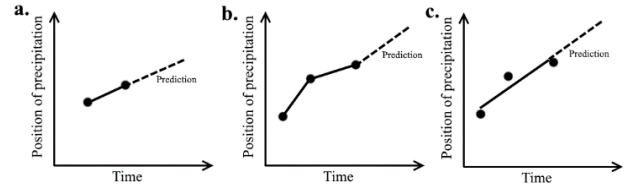


Fig. 2. The extrapolation of precipitation position by the basic translation algorithm (a), step translation algorithm (b) and sequence translation algorithm (c).

In this section precipitation prediction algorithms, used in this research, will be presented. Translation algorithms work by extracting precipitation movement vector from consequent weather radar images and extrapolating them into the future. These algorithms differ in the way how a movement vector is extracted.

The CNN-based algorithm uses machine learning techniques to predict subsequent weather radar images in the future.

##### A. Persistency

Persistency algorithm is an assumption that all the initial conditions will remain stable in the future. This means that this algorithm returns the initial weather radar image for every period of a forecast. This technique is commonly used in weather prediction accuracy evaluations as a benchmark. If an accuracy of a weather prediction algorithm is lower than the one with persistency assumption, the quality of an algorithm is poor.

##### B. Basic translation algorithm

This basic translation algorithm takes two consequent weather radar images and finds an anticipated precipitation movement vector between them. Using this vector, an arbitrary amount of radar images can be generated by performing an image translation at each forecast step (Figure 2, a).

The algorithm uses brute force to find a horizontal and a vertical pixel offset at which a correlation value between the two images is the highest. A Hanssen–Kuipers score, defined in the fourth chapter, is used as a correlation value.

##### C. Step translation algorithm

It might not always be possible to find an accurate precipitation movement vector from just the two consequent images. Furthermore, movement vector can only have integer values. These problems are addressed with a step translation algorithm.

This algorithm takes three subsequent weather images and computes the best movement vector for each adjacent pair of two images with the same method as in the basic translation

algorithm. Then, an average of these vectors is obtained and used as the final best movement vector, from which the forecast images are generated. (Fig. 2, b.)

Since the obtained average vector can have non-integer values (and it is impossible to move an image with a non-integer offset of pixels without using additional transformation), both source image and vector itself are scaled up by the same factor to perform translation with integer values. After this process, the image is resized down to its original resolution.

#### D. Sequence translation algorithm

Although step translation algorithm ensures that a precipitation movement vector is obtained more accurately, there still might be errors while determining movement direction between two weather radar images.

The sequence algorithm, same as the step translation algorithm, uses three radar images to determine the direction of precipitation, but this algorithm computes the best movement vector for the whole sequence at once (Figure 2, c). Sequence translation algorithm computes a sum of the HK scores for each pair of adjacent images at every possible translation vector value. The best movement vector is determined by the highest sum of the HK scores.

#### E. CNN-based algorithm

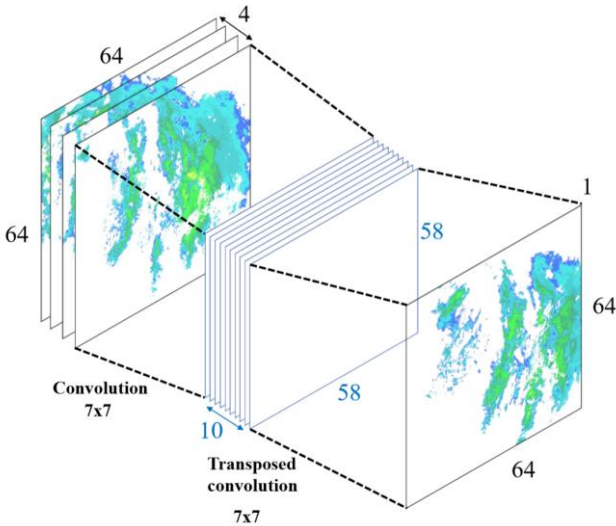


Fig. 3. The architecture of CNN-based prediction algorithm. Network uses convolution and transposed convolution to predict the next image from four previous weather radar images.

This algorithm is based on an architecture of convolutional neural network (Figure 3). It consists of three layers of neurons.

Input layer receives four subsequent weather radar images with a resolution of 64 x 64. Each image in a sequence is represented as a different channel of an input (similarly to how RGB color channels are represented in an ordinary image).

Next, convolution is applied between the input and the hidden layer, using a kernel with a size of 7 x 7, which reduces

the resolution of the images in the hidden layer to 58 x 58. The number of channels in the hidden layer is expanded to 10. The kernel size of 7 x 7 was selected to capture the possible movement of precipitation between the first and the fourth input images in a single kernel. 10 channels in the hidden layer yielded the best results during the experiments.

Finally, a transposed convolution (sometimes called deconvolution) is applied between the hidden and the output layer with a single channel. This transforms an image into the original resolution of 64 x 64. Resulting image is an output of a neural network and represents generated map of precipitation for the next time step after four input images.

The architecture of this neural network can only predict a single weather radar image into the future. To generate an arbitrary amount of result images, each output of the network is passed into the input of the next iteration, which generates precipitation image for the subsequent time step.

## VI. SINGLE EXPERIMENT RESULTS

A precipitation event of 29/10/2017 was selected to compare the results of the prediction algorithms. Weather radar image obtained at 09:45 AM local time, together with three previous images, was used as a source image. Every algorithm predicted two hours of precipitation into the future.

Figure 4 displays a forecasted precipitation with all the algorithms mentioned above, together with actual observed conditions. An HK score, that determines the accuracy of every forecast when compared with the actual precipitation, is displayed under each predicted image.

For this precipitation event, the CNN-based algorithm outperformed every other algorithm, including persistency benchmark. Its HK score was the highest at almost every step of the forecast. In fact, for this particular event, only Basic translation algorithm failed to outperform persistency benchmark.

Every algorithm obtained different best precipitation movement vector: a pixel offset of [0, 1] was obtained by the basic translation algorithm, [0.25, 0.5] by the step translation algorithm and [1, 0] by the sequence translation algorithm. Positive x values indicate movement to the east and positive y values to the south.

Translation algorithms try to predict only the movement of precipitation, without considering changes in strength and shape, but can still yield reasonably accurate results for the first hour of the forecast. On the other hand, CNN-based algorithm managed to predict that precipitation system will rotate counter-clockwise during this particular event and maintained rather accurate evaluation of possible precipitation strength in the area.

However, the CNN-based algorithm has lost some precipitation shape details during the longer forecast and predicted rather smooth contour in contrast to the actual more scattered shape.

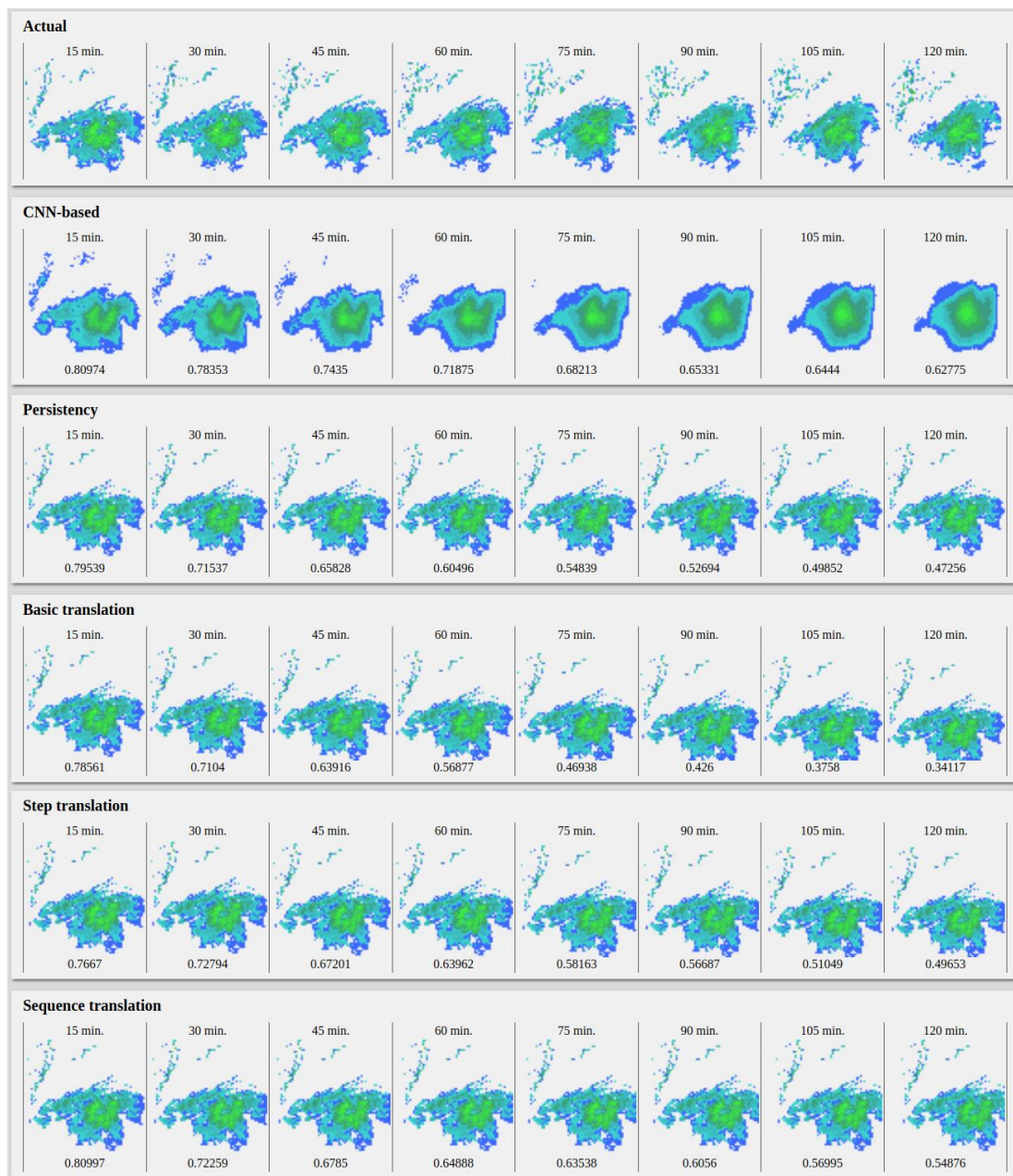


Fig. 4. The algorithm results for a precipitation event of 29/10/2017 09:45 AM. Each row consists of a set of images for up to 2 hours with 15 minutes intervals, generated by each algorithm. The number below each image indicates an evaluated HK accuracy score.

Base translation algorithm failed to obtain a correct direction to which precipitation was moving. This indicates that two subsequent images are not always enough to correctly calculate the movement direction.

## VII. AVERAGED RESULTS

To compare the accuracy of the algorithms for a longer period, a week of 23/10/2017 to 30/10/2017 was chosen. Every algorithm generated 8 weather prediction images for two hours into the future at 15 minutes intervals. Generated images were compared with an actual precipitation to obtain an HK for each pair of the images. Then, the average scores for every step of the

forecast were calculated. The comparison of an accuracy of the algorithms is displayed in Figure 5.

Comparison results show that CNN-based algorithm outperforms every other algorithm for almost two hours of the forecast. Sequence translation algorithm was the most accurate among the extrapolation algorithms and exceeded precipitation benchmark for the first 90 minutes of the forecast.

Step translation and Basic translation algorithms performed poorly. Their accuracy was much lower than the persistency benchmark score for the forecasts longer than 30 minutes.

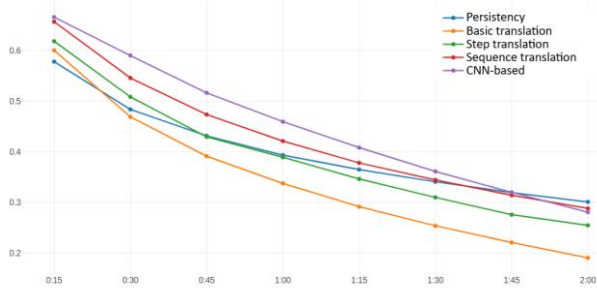


Fig. 5. The comparison of the HK score of the algorithms for a week of 23/10/2017 to 30/10/2017. x axis defines the duration of the forecast; y axis – HK prediction accuracy score.

Nonetheless, the prediction accuracy of every algorithm decreases rapidly, and the only CNN-based algorithm has an accuracy higher than 0.5 at 45 minutes forecast. However, as explained in the fourth chapter, selected method of HK score evaluation does not include additional scores for near misses, when precipitation is predicted correctly with an offset of a few pixels.

### VIII. CONCLUSIONS AND FUTURE WORK

In this research, it was shown that precipitation movement extrapolation algorithm can outperform persistency benchmark if movement direction is obtained correctly. In addition to this, even a simple convolutional neural network can predict movement and changes in precipitation shape reasonably well. However, the accuracy of a prediction decreases rapidly and can't be trustworthy for periods longer than an hour.

Presented precipitation translation algorithms are very simple and do not take the precipitation system rotation and rain strength into account. The extrapolation of these additional values may help to increase prediction accuracy.

Furthermore, although CNN-based algorithm performed the best, is not the most suitable machine learning algorithm to predict changes in time, since it has no memory of the previous inputs, which might be important when predicting precipitation further into the future. There are better neural network architectures to tackle this problem, like Recurrent Neural Networks (RNN) or Long Short-Term memory (LSTM) networks. In addition to this, movement vectors obtained with

extrapolation techniques can be used as additional features to improve machine learning accuracy.

Finally, the accuracy of official precipitation forecasts should also be evaluated to better understand how extrapolation and machine learning prediction accuracy compares to numerical forecasts.

This research is still at a very early stage and presents only the basic algorithms, however, a broad spectrum of available techniques in this area (such as the inclusion of precipitation system rotation and rain strength extrapolation, or various more sophisticated machine learning methods for prediction of time series) will allow further improvements in the forecast accuracy and duration.

### REFERENCES

- [1] Harvey Stern, Noel E. Davidson, „Trends in the skill of weather prediction at lead times of 1–14 days,“ *Quarterly Journal of the Royal Meteorological Society* banner, pp. 2726-2736, 2015.
- [2] L. Lia, W. Schmida, J. Jossb, „Nowcasting of Motion and Growth of Precipitation with Radar over a Complex Orography,“ *Journal of applied meteorology and climatology*, June 1995.
- [3] Saxen, Mueller, Warner, Steiner, Ellison, Hatfield, Betancourt, Dettling, Oien, „The operational mesogamma-scale analysis and forecast system of the U.S. Army Test and Evaluation Command. Part IV: The White Sands Missile Range auto-nowcast system,“ *Journal of Applied Meteorology and Climatology*, April 2008.
- [4] Li, Lai, „Applications of radar-based nowcasting techniques for mesoscale weather forecasting in Hong Kong,“ *Meteorological applications*, pp. 253-264, 2004.
- [5] James W. Wilson, N. Andrew Crook, Cynthia K. Mueller, Juanzhen Sun, and Michael Dixon, „Nowcasting Thunderstorms: A status report,“ *Bulletin of the American Meteorological Society*, pp. 2079-2099, 1998.
- [6] Holmstrom, Liu, Vo, „Machine Learning Applied to Weather Forecasting,“ 15 December 2016.
- [7] Campolo, Andreussi, Soldati, „River flood forecasting with a neural network model,“ *Water resources research*, pp. 1191-1197, April 1999.
- [8] Denoux, Rizand, „Analysis of Radar Images for Rainfall Forecasting using Neural Networks,“ *Neural computing and applications*, pp. 50-61, 1995.
- [9] Richard. J. Vogt, „A way forward windfarm - weather radar coexistence“.
- [10] Gauthreaux, „Weather Radar Quantification of Bird Migration,“ *BioScience*, pp. 17-19, 1 January 1970.
- [11] John L. McBride, Elizabeth E. Ebert, „Verification of Quantitative Precipitation Forecasts from Operational Numerical Weather Prediction Models over Australia,“ *Weather and Forecasting*, pp. 103-121, 2000.