**Shape

Description automatically generated with medium confidence**

**Kauno technologijos universitetas**

Elektros ir elektronikos fakultetas

**Skaitinio intelekto metodai (T125B117)**

Kursinis darbas

**Nekalbių garsų klasifikavimas naudojant konvoliucinį neuroninį tinklą**

|  |
| --- |
|  |
| **Žygimantas Marma EVS 8/1**  Studentas |
|  |
| **Prof. Vidas Raudonis**  Dėstytojas |
|  |

**Kaunas, 2022**

**Turinys**

Contents

[**Turinys** 2](#_Toc91371610)

[Įvadas 4](#_Toc91371611)

[1. Literatūros apžvalga 6](#_Toc91371612)

[1.1 Music Genre Classification (Derek A. Huang) 6](#_Toc91371613)

[1.2 Classifying environmental sounds using image recognition networks (Venkatesh Boddapatia) 6](#_Toc91371614)

[1.3 Development of sound identification system for domestic actions recognition (Arūnas Lipnickas) 7](#_Toc91371615)

[1.4 Darbo pavadinimas 8](#_Toc91371616)

[1.5 Darbo pavadinimas 8](#_Toc91371617)

[2. Matematinis siūlomo sprendimo pagrindimas 8](#_Toc91371618)

[2.1 Analizuojamas duomenų rinkinys 8](#_Toc91371619)

[2.2 Bruožų išgavimas iš audio failo 10](#_Toc91371620)

[2.3 Duomenų klasifikavimo metodas 12](#_Toc91371621)

[2.3.1 Dvimatis konvoliucinis sluoksnis (Conv2D) 14](#_Toc91371622)

[2.3.2 Parametrų išmetimo technika (Dropout) 15](#_Toc91371623)

[2.3.3 Partijos normalizavimas (Batch Normalization) 16](#_Toc91371624)

[2.3.4 Optimizat 17](#_Toc91371625)

[3. Testavimas 17](#_Toc91371626)

[Išvados 18](#_Toc91371627)

[Naudotos literatūros sąrašas. 18](#_Toc91371628)

**Santrumpų ir terminų sąrašas**

**Santrumpos:**

Angl. ESC: Dataset for Environmental Sound Classification

MFCC - Mel Frequency Cepstral Coefficients

CRP - Cross Recurrence Plot

**Terminai:**

**Fonema** ([sen. gr.](https://lt.wikipedia.org/wiki/Senov%C4%97s_graik%C5%B3_kalba) φώνημα 'garsas') – mažiausias [kalbos](https://lt.wikipedia.org/wiki/Kalba) vienetas, turintis skiriamąją reikšmę.

[**1 pav.** Mokslinio darbo testavimo rezultatai 6](#_Toc91459350)

[**2 pav.** Darbe naudojamas trijų tipų vaizdų derinys, norint gauti vienos spalvos vaizdą. 7](#_Toc91459351)

[**3 pav.** Darbe naudota neuroninio tinklo struktūra 7](#_Toc91459352)

[**4 pav**. Urbansound8k duomenų rinkinio taksonomijos iškarpa 10](#_Toc91459353)

[**5 pav.** Funkcija nubrėžiame signlo amplitudės priklausomybes nuo laiko 11](#_Toc91459354)

# Įvadas

* *Įvadas (kokia problema sprendžiama, kokia nauda visuomenei, ar yra rinka siūlomam sprendimui);*

Gyvendami pasaulyje, apsuptame skirtingų garsų iš skirtingų šaltinių. Mūsų smegenys kartu su klausos sistema nuolat dirba identifikuojant kiekvieną girdimą garsą evoliucija pragrįstu optimaliu būdu. Be to, mūsų smegenys nuolat apdoroja gautus garso signalus ir suteikia mums atitinkamų žinių apie supančią aplinką. Akivaizdu, kad žmonės gali lengvai atskirti garsus, tačiau kompiuterizuotoms sistemoms ši užduotis nėra tokia paprasta. Nors buvo bandymų skirtingų algoritmų pagalba sukurti išmaniuosius įrenginius, kurie galėtų išgauti reikiamą informaciją iš garso signalo, tačiau smegenų tikslumo lygio pasiekti mokslininkams vis dar nepavyksta. Būtent pastarąją garsų klasifikavimo problemą ir nagrinėsime šiame darbe.

Garso klasifikavimas – tai garso įrašų klausymosi ir analizės procesas. Šis procesas, taip pat žinomas kaip audio signalų klasifikavimas, yra pagrindas daugelių šiuolaikinių DI technologijų, tokių kaip: virtualieji asistentai, automatinės kalbos atpažinimo sistemos ir teksto į kalbą vertimo aplikacijos. Garsų klasifikavimas jau daugelį metų yra didelės svarbos tyrimų sritis. Šioje srityje buvo išbandyta daug įvairių metodų su skirtingais modeliais ir funkcijomis, kurie pasirodė esą naudingi ir tikslūs. Taip pat garsų klasifikavimą galima rasti išmaniųjų namų apsaugos sistemose, realiu laiku kalbos vertimo iš vienos į kitą, bioakustinės stebėsenos, įsibrovėlių aptikimo laukinės gamtos zonose ir garso stebėjimo srityse.

Aplinkos garsų klasifikavimas yra viena iš svarbiausių problemų garsų atpažinimo srityje. Palyginus su įprastais ir struktūriškais garsais, tokiais kaip kalba ar muzika, aplinkos garsai neturi nei statinių laiko modelių(static time patterns), kaip melodijos ar ritmai, nei semantinių sekų, kaip fonemos. Todėl sunku rasti universalių bruožų, galinčių reprezentuoti įvairius temprų modelius. Be to, aplinkos garsuose yra daug triukšmo ir pašalinių garsų nesusijusių su nagrinėjamu. To pasėkoje susidaro sudėtinga kompozicijos struktūra su nepastovumu, įvairumais ir nestruktūrizuotomis savybėmis. Siekiant išspręsti aukščiau išvardytas problemas, nekalbių aplinkos signalų (ESC) užduotims atlikti buvo naudojami įvairūs signalų apdorojimo metodai ir mašininio mokymosi metodai.

Pastaraisiais metais aplinkos garso klasifikavimo (ESC) tyrimams, kurie daugiausia skirti tokiems garso įvykiams kaip šunų lojimo, šūvių ar oro kondicionavimo garsų atpažinimui, sulaukia vis daugiau dėmesio. Tyrimo rezultatai buvo panaudoti daugelyje praktinių pritaikymų, įskaitant robotizuotą klausą, išmaniuosius namus, garso stebėjimo sistemą garso kraštovaizdžio vertinimą.

Modeliai taip pat naudojami apsaugos sistemose aptikti tokius garsus kaip dūžtantį stiklą. Pramonėje naudojamas nuspėjamai priežiūrai, nustatant garso neatitikimus gamyklinėse mašinose. Jis netgi naudojamas atskirti gyvūnų raginimus stebėti ir saugoti laukinę gamtą. PARAŠYTI KAD MEDICINOJE ROBOTAI NAUDOJAMI SLAUGAI

Svarbu paminėti kad, projektų, susijusių su garso klasifikavimu, duomenų rinkinio kokybė gali nulemti ir dažniausiai nulemia projekto rezultatų kokybę. Todėl, norint užtikrinti tikslų garso klasifikavimo lygį, mums reikės daug aukštos kokybės, tiksliai anotuotų duomenų. Galiausiai eksperimentai su dviem viešais duomenų rinkiniais – UrbanSound 8 K ir ESC-50 – rodo siūlomo metodo veiksmingumą.

Galiausiai šiame projekte mes pasiūlysime konvoliucinio neuroninio tinklo modelį (CNN), kurio dėka būtų galima klasifikuoti kalbius garso signalus naudojant Mel-Spectograms.

# 1. Literatūros apžvalga

* Literatūros apžvalga (min 5 straipsniai, kuriuose sprendžiama ta pati problema, kaip tuose straipsniuose yra sprendžiama);

## Music Genre Classification (Derek A. Huang)

Darbe mokslininkai klasifikavo muzikos žanrus įvairiais metodais. Buvo atlikti eksperimentai su RBF branduolio atraminių vektorių mašina (angl. RBF Kernel Support Vector Machines), k-artimiausių kaimynų metodu, pagrindinio persiuntimo (angl. basic feed-forward) tinklu ir galiausiai pažangiu konvoliuciniu neuroniniu tinklu.

Treniravimui buvo naudotas GTZAN duomenų rinkinys. Šis duomenų rinkinys susideda iš 1000 30 sekundžių garso klipų, kurie visi yra pažymėti kaip vienas iš 10 galimų žanrų ir pateikti kaip .au failai. Mokslininkai taip pat naudojo Mel‘o spektogramas duomenų analizei. Galiausiai buvo prieita išvados, kad tinkamiausias yra konvoliucinis neuroninis tinklas. Šio tipo tinklas pasiekė 82% tikslumą su testavimo duomenimis (naudojant apdorotus duomenis), kai kiti metodai pasiekė 54% bei 60% tikslumą.

Table

Description automatically generated

**1 pav.** Mokslinio darbo testavimo rezultatai

## Classifying environmental sounds using image recognition networks (Venkatesh Boddapatia)

Darbe yra naudojami konvoliuciniai gilieji neuroniniai tinklai (,,AlexNet“ ir ,,GoogLeNet“) vaizdų atpažinimui nekalbių signalų klasifikavimui. Buvo dirbama su ESC-10, ESC-50 ir UrbanSound8K garsų duomenų rinkiniais. Kadangi ESC duomenų bazės yra ganėtinai mažos (ESC-50 sudaryta iš 2000 5s įrašų, ESC-10 iš xxx) todėl mokymosi duomenų bazė buvo praplėsta. Nauji garso įrašai, gauti modifikuojant pradinius duomenis įvairiais metodais: įtraukiant atsitiktinius vėlavimus, audio signalo laikas prailginamas, garso įrašų pakeičiamas greitis – jie pagreitinami arba sulėtinami tam tikru koeficientu. Šiame darbe yra palyginamos skirtingos vaizdiniai rūšys (spektrogramos, Mel dažnio cepstraliniai koeficientai (MFCC) ir kryžminiai pasikartojimo grafikai (CRP)) siekiant naudojant vaizdus gauti geriausią garsinių signalų klasifikavimo tikslumą.

Mokslininkai įrodė, kad gilieji konvoliuciniai neuroniniai tinklai, specialiai sukurti objektų atpažinimui vaizduose, gali būti sėkmingai išmokyti klasifikuoti aplinkos garsų spektrinius vaizdus. Geriausias klasifikavimo tikslumas ESC-50, ESC -10 ir UrbanSound8K duomenų rinkiniuose buvo atitinkamai 73 %, 91 % ir 93 % naudojant GoogLeNet. Daugeliu atvejų, kuriuos ištyrė mokslininkai, „GoogLeNet“ tinklas klasifikavimo tikslumas buvo didesnis nei „AlexNet“. Jų manymų to pagrindinė priežastis yra ta, kad „GoogLeNet“ yra daug gilesnis nei „AlexNet“ (22, palyginti su 8 sluoksniais). Taip pat mokslininkai nustatė, kad kelių skirtingų vaizdo rūšių kombinavimas (spektogramos, MFCC, ir CRP) kaip to paties vaizdo skirtingų spalvų kanalų nepagerino klasifikavimo tikslumo.

A picture containing text

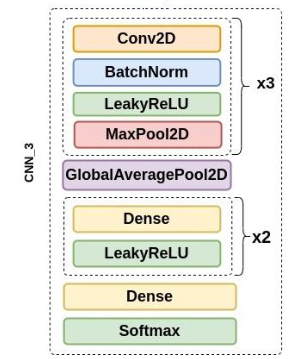
Description automatically generated

**2 pav.** Darbe naudojamas trijų tipų vaizdų derinys, norint gauti vienos spalvos vaizdą.

## Development of sound identification system for domestic actions recognition (Arūnas Lipnickas)

Šiame darbe yra pateikiamas garso įrašų apdorojimo metodas, kuris suskirsto aplinkos garsus į vieną iš 11 buitinės veiklos klasių. Mokslininkų aprašytas metodas pagrįstas bruožų išskyrimu iš Mel’o kepstrumo į 224 pikselių pilkos spalvos kvadratinį vaizdą. Šie vaizdai buvo klasifikuojami naudojant 3 sluoksnių konvoliucinį neuroninį tinklą.

Geriausiu atveju buvo pasiektas 92,60 % atpažinimo tikslumas naudojant DASEE duomenų bazę. Šie rezultatai pranoko 1 dimensijos konvoliuciniu modeliu pagrįstus metodus, kuriuose naudojami neapdoroti garso signalo duomenys. DASEE duomenų bazės tarpklasių disbalanso problema buvo išspręstas taikant klasės kodavimo etikečių išlyginimą (angl. labels smoothing) ir klasių svėrimą kategoriškoje kryžminės entropijos (angl. categorical cross-entropy) praradimo funkcijoje.



**3 pav.** Darbe naudota neuroninio tinklo struktūra

## Rethinking the Usage of Batch Normalization and Dropout in the Training of Deep Neural Networks (Guangyong Chen)

Basasd adsas da sd

In this work, we propose a novel technique to boost training efficiency of a neural network. Our work is based on an excellent idea that whitening the inputs of neural networks can achieve a fast convergence speed

For example, we should not place Batch Normalization before ReLU since the non- negative responses of ReLU will make the weight layer updated in a suboptimal way, and we can achieve better performance by combining Batch Normalization and Dropout together as an IC layer.

4.2. CIFAR10/100

However, as shown in Fig. 4, the traditional usage of Batch-  
Norm still leads to an unstable optimization process com-  
pared with our implementation, which inspires us to ques-  
tion the common practice of placing BatchNorm before  
the activation layer.

In this paper, we rethink the usage of BatchNorm and  
Dropout in the training of DNNs, and find that they should  
be combined together as the IC layer to transform the activa-  
tion into independent components, and place this layer right  
before the weight layer.

By reformulating ResNets with IC layers, we  
achieve more stable training process, faster convergence  
speed and better generalization performance. In future, we  
should consider incorporating more advanced normaliza-  
tion methods, such as layer normalization (Ba et al., 2016),  
instance normalization (Ulyanov et al., 2016), group nor-  
malization (Wu & He, 2018) etc., into the IC layer, and  
consider more advanced statistical techniques to construct  
independent components.

## Environmental sound classification using temporal‑frequency attention based convolutional neural network (Wenjie Mu)

Basasd adsas da sd

# 2. Matematinis siūlomo sprendimo pagrindimas

Matematinis siūlomo sprendimo pagrindimas (algoritmai, pseudo kodai, formulės, viskas kas paaiškina kaip veikia jūsų pasirinktas metodas);

## 2.1 Analizuojamas duomenų rinkinys

Aplinkos signalams klasifikuoti, kaip minėjome anksčiau, naudosime UrbanSound8K duomenų bazę. Šiame duomenų rinkinyje yra 8732 pažymėtos miesto aplinkos garsų ištraukos (kurių trukmė <=4s) iš 10 klasių: oro kondicionierius, automobilio garsinis signalas, vaikų žaidimo šurmulys, šuns lojimas, gręžimas, variklio darbas tuščiąja eiga, ginklo šūvis, jackhammer, sirenos garsas ir gatvės muzika. Klasės sudarytos iš miesto garso taksonomijos 2.1.1 pav. Išsamų duomenų rinkinio ir jo sudarymo aprašymą rasti moksliniame darbe[6].

Diagram

Description automatically generated

**4 pav**. Urbansound8k duomenų rinkinio taksonomijos iškarpa

Rinkinyje visos ištraukos paimtos iš įrašų, įkeltų į [www.freesound.org](http://www.freesound.org) tinklalapį. Failai yra iš anksto surūšiuoti į dešimt aplankalų (aplankai, pavadinti fold1-fold10), kad būtų lengviau atkurti ir palyginti su automatinio klasifikavimo rezultatais, aprašytais aukščiau minėtame straipsnyje[6]. Duomenų rinkinyje taip pat pateikiami metaduomenys (2.1 lentelė) kiekvienam audio įrašui, kurių pagalba galime naudotis esamais garso įrašais.

2.1 lentelė. Urbansound8k metaduomenų fragmentas

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **slice\_file\_name** | **fsID** | **start** | **end** | **salience** | **fold** | **classID** | **class** |
| 100032-3-0-0.wav | 100032 | 0 | 0.317551 | 1 | 5 | 3 | dog\_bark |
| 100263-2-0-117.wav | 100263 | 58.5 | 62.5 | 1 | 5 | 2 | children\_playing |
| 100263-2-0-121.wav | 100263 | 60.5 | 64.5 | 1 | 5 | 2 | children\_playing |
| 100263-2-0-126.wav | 100263 | 63 | 67 | 1 | 5 | 2 | children\_playing |
| 100263-2-0-137.wav | 100263 | 68.5 | 72.5 | 1 | 5 | 2 | children\_playing |
| 100263-2-0-143.wav | 100263 | 71.5 | 75.5 | 1 | 5 | 2 | children\_playing |
| 100263-2-0-161.wav | 100263 | 80.5 | 84.5 | 1 | 5 | 2 | children\_playing |

Norėdami pamatyti savo audio siganalus grafine formą pasirašome python funkciją. Naudodami „Librosa“ biblioteką pasikiame diske esančius wav formato failus ir nubrėžiame signlo amplitudės priklausomybes nuo laiko dviem atsitiktiniam įrašam.

Text

Description automatically generated

**5 pav.** Funkcija nubrėžiame signlo amplitudės priklausomybes nuo laiko

Chart

Description automatically generated

**6 pav.** Audio signalų amplitudės priklausomybes nuo laiko, viršuje šuns lojimas, apačioje mašinos variklio ūžimas

## 2.2 Bruožų išgavimas iš audio failo

Yra 3 pagrindiniai būdai, kaip išgauti funkcijas iš garso failo:

1. Naudojant garso failų mffcs duomenis (Panašiai kaip music genres)[7]
2. Garso spektrogramų vaizdo naudojimas. Turint vaizdus vienas iš pranašumų kuriant neuroninį tinklą yra tai, kad galima naudoti konvoliucinius sluoksnius.
3. Naudoti keletą bruožų išgavimo technikų ir jas kombinuoti.

Šiame darbe nagrinėsime garsų apžinimą naudojantis Mel sprektogramomis (angl. Mel spectrogram).

Mel‘o skalė yra suvokimo skalė natų (pitch), kuriuos klausytojai vertina kaip vienodus atstumus vienas nuo kito. Ši skalė buvo įvesta, kadangi žmonių klausa yra jautresnė garsų pasikeitimui žemų dažnių zonoje nei aukštųjų. Atskaitos taškas tarp šios skalės ir normalaus dažnio matavimo nustatomas priskiriant 1000 mels suvokimo aukštį 1000 Hz tonui, 40 dB virš klausytojo slenksčio. Viršijus 500 Hz, vis didesni intervalai klausytojų vertinami taip, kad tolygiai padidėtų. Nors oficialios formulės pakeistį dažnį į mel‘us nėra, populiariausiai pripažinta yra formulė iš O'Shaughnessy's knygos: [idėti linką]

Šią išraišką atvaizduojame grafiškai 2.2.1 pav.

Chart

Description automatically generated

**2.2.1 pav**. Pikio melo skalės ir Hertzo skalės diagramos

Suprasti kokie vaizdai susidaro garso signalui pritaikant Mel‘o spektogramas parašome Python funkciją:

Text, letter

Description automatically generated

**2.2.1 pav**. KAZKOKIA FKNC!

A picture containing chart

Description automatically generated

**2.2.2 pav**. Audio signalas Mel‘o skalėje

## 2.3 Duomenų klasifikavimo metodas

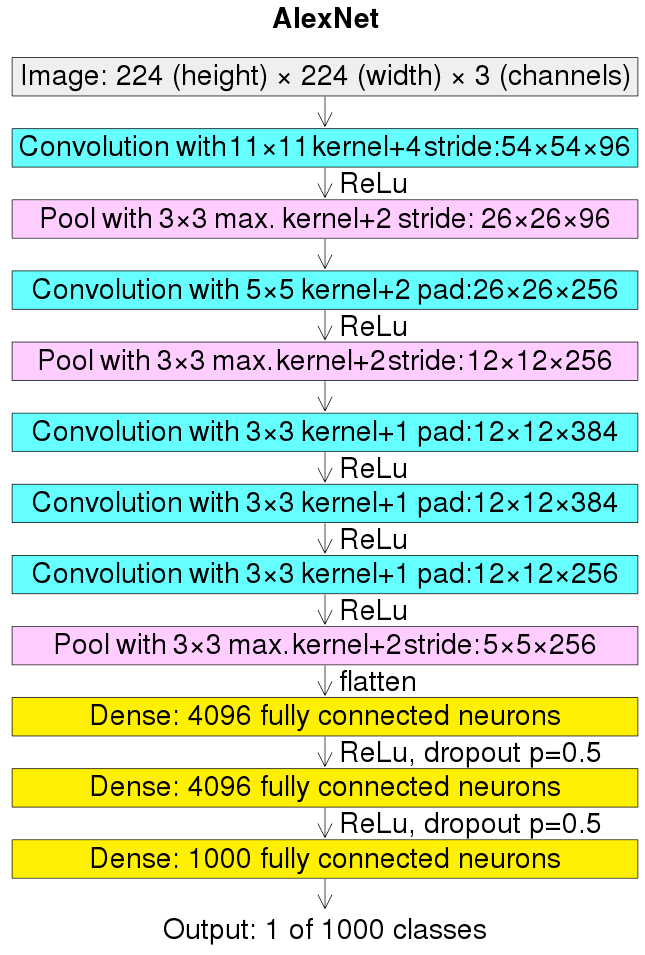
Kaip minėjome anksčiau duomenų klasifikavimui naudosime konvoliucinį neuroninį tinklą. Šio tipo dirbtiniai neuroniniai tinklai, dažniausiai taikomi kompiuterinės regos srityje: vaizdų atpažinimui, objektų radimui ir identifikavimui nuotraukose, vaizdų segmentacijai ir natūralios kalbos atpažinimui. Pats pavadinimas „konvoliucinis neuroninis tinklas“ rodo, kad tinklui mokantis naudojama matematinė operacija, vadinama konvoliucija. Šie tinklai yra specializuotas neuroninių tinklų tipas, kuriame konvoliucija naudojama vietoj bendros matricos daugybos bent viename iš jų sluoksnių.[wiki] Tuo tarpu konvoliucinis neuroninis tinklas susideda iš įėjimo sluoksnio, paslėptų sluoksnių ir išėjimo sluoksnio.

Chart

Description automatically generated

**2.3.1 pav**. Tipinė konvoliucinio neuroninio tinklo struktūra

Šiame darbe naudosime tinklo struktūrą panašią į AlexNet tinklą.[X] Pastarąją konvoliucinio neuroninio tinklo (CNN) architektūrą, kurią sukūrė Alex Krizhevsky, bendradarbiaudamas su kitais mokslininkais. 2012 m. rugsėji AlexNet dalyvavo „ImageNet Large Scale Visual Recognition Challenge“ varžybose.[3] Tinklas buvo tarp 5 geriausių darbų pagal klaidų lygį - 15,3 %, o antroje vietoje likusį modelį aplenkė daugiau nei 10,8 procentinio punkto. Pagrindinis pirminio dokumento rezultatas buvo tai, kad modelio gylis buvo labai svarbus jo aukštam našumui. Vis dėlto didelis parametrų skaičius, kuris buvo brangus skaičiavimo požiūriu, tačiau buvo įmanomas dėl grafikos apdorojimo blokų (GPU) naudojimo mokymo metu. Eksperimentai buvo atlikti 2012m. todėl šis modelis buvo išskirstytas į dvi dalis, kurios paraleliai dirbo naudojant dvi vaizdo plokštes (GPU). Ši problema mums nebėra aktuali, nes dabartinės įrangos galimybės leidžia treniruoti sudėtingų struktūrų modelius su milijonais parametrų naudojantis paprastas žaidimams skirtas vaizdo plokštes.



**10 pav**. AlexNet konvoliucinio neuroninio tinklo struktūra

Naudodami konvoliucinius sluoksniu, MaxPooling2D ir Dropout funkcijas sudarome savo neuroninį tinklą kurio įėjimas būtų spektogramos vaizdas(64x128 pikselių), o išėjimas klasės kuriai priskiriamas garsas numeris. Tinklo struktūrą pateikiame 10 pav.

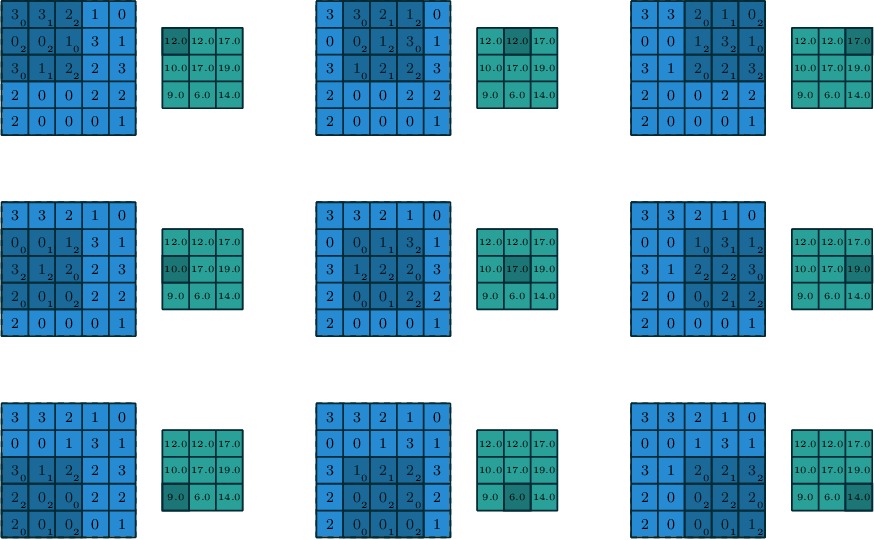
A picture containing calendar

Description automatically generated

**11 pav**. Pradinė tinklo struktūra

## 2.3.1 Dvimatis konvoliucinis sluoksnis (Conv2D)

Dvimatė konvoliucija yra gana paprasta operacija: sukuriame branduolį (angl. kernel), kuris yra tiesiog maža svorių matrica. Šis branduolys „slenka“ per 2D įvesties duomenis, atlikdamas elementų dauginimą iš įvesties dalies, kurioje jis šiuo metu yra, tada rezultatai yra sumuojami į vieną išvesties pikselį. Branduolys pakartoja šį procesą kiekvienoje vietoje, per kurią slysta, konvertuodamas 2D funkcijų matricą į dar vieną 2D reikšmių matricą. Išėjimo reikšmės iš esmės yra įėjimo reikšmių svertinės sumos, esančios maždaug toje pačioje įvesties sluoksnio išėjimo pikselio vietoje.



**11 pav**. Dvimatės konloviucijos pavyzdys

Kaip matome iš 11 pav. dvimatė konvoliucija sumažina pradinį vaizdą. Paprasčiausiu atveju sluoksnio išėjimo reikšmę, kai įvesties dimensijos yra (N, Cin, H, W), o išvesties (N, Cout, Hout, Wout), galima tiksliai apibūdinti taip:

Diagram, schematic

Description automatically generated

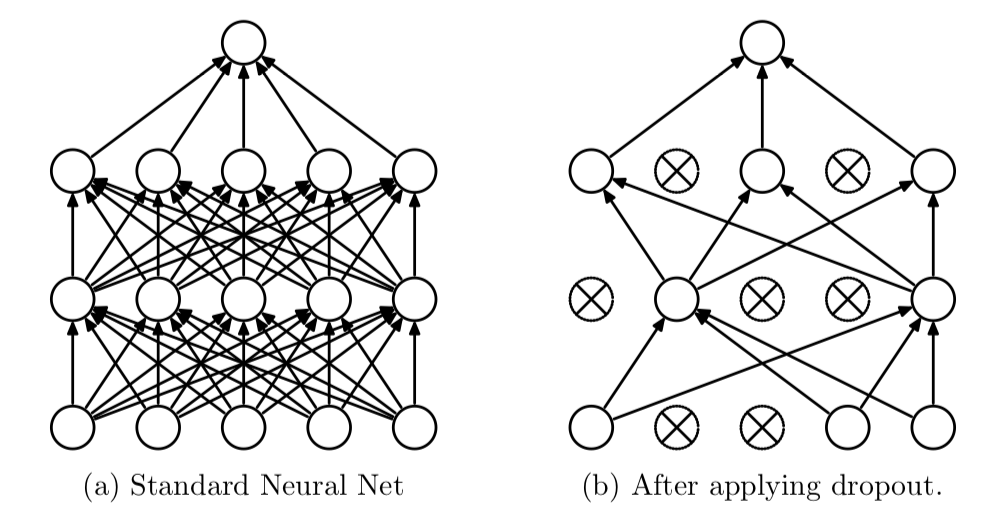
Kur ★ yra galiojantis 2D kryžminės koreliacijos operatorius, N yra paketo dydis (angl. batch size), C reiškia kanalų skaičių, H yra įvesties vaizdo aukštis pikseliais, o W yra plotis pikseliais. Šį sluoksnį galime naudoti pasitelkiant TensorFlow esančias funkcijas: *tf.keras.layers.Conv2D*

Sluoksnio aktyvacijos funkcijai naudosime ReLU (angl. Rectified Linear Unit). Kompiuterinėje regoje, siekiant duomenų normalizacijos, populiariausia yra ReLU aktyvavimo funkcija, kuri yra įvardijama, kaip geriausiai tinkanti neuroninio tinklo mokymo efektyvumui. Ši funkcija aprašoma kaip:

Akivaizdu kad ši funkcija gautas neigiamas įėjimo reikšmės, po konvoliucijos operacijos, pakeičia į 0.

## 2.3.2 Parametrų išmetimo technika (Dropout)

Išmetimas (angl. Dropout) yra reguliarumo metodas, skirtas sumažinti persimokymą ir pagerinti giliųjų neuronų tinklų testavimo rezultatus. Išmetimas reiškia neuronų (paslėpto ar matomo sluoksnio) išmetimą neuroniniame tinkle. Išmetimą galime interpretuoti kaip tikimybę, kad tam tikras sluoksnio neuronas bus paliktas mokymuisi. Kur 1,0 reiškia, kad nėra iškritimo, o 0,5 reiškia, kad 50% sluoksnio neuronų yra ignoruojami. Akivaizdu kad tada, kiekviena iteracija naudoja skirtingą modelio parametrų pavyzdį, o tai verčia kiekvieną neuroną turėti tvirtesnių(robust features) savybių, kurias galima naudoti su kitais atsitiktiniais neuronais. Nors naudojant šią techniką yra mažinamas neuroninio tinklo sudėtingumas (sumažėja apmokamų parametrų skaičius), tačiau išmetimas taip pat padidina mokymo laiką, reikalingą modelio konvergencijai. Apibendrinus išmetimas yra puiki technika stengiantis sumažinti tinko prisitaikymą prie testavimo duomenų ir padidinti gebėjimą teisingai klasifikuoti nematytus duomenis. Todėl remdamiesi AlexNet struktūra (kurioje naudoti du išmetimo sluoksniai su 0,5 tikimybę) į savo modelį įvesime analogišką techniką į du priešpaskutinius sluoksnius, o viename sluoksnyje arčiau įvedimo duomenų naudosime išmetimą su 0,1 tikimybę.



**11 pav**. Įprastas neuroninis tinklas (kairėje) ir tinklas naudojant išmetimo techniką (dešinėje)

## 2.3.3 Partijos normalizavimas (Batch Normalization)

Ideti images kuru norm vs kur ner

Chart, histogram

Description automatically generatedChart, line chart

Description automatically generated

A picture containing text, indoor

Description automatically generated

## 2.3.4 Optimizat

Opimizatorius calbackai

# 3. Testavimas

* Testavimas (kažkokia grafinė medžiaga parodanti kaip veikia jūsų programa);

Ikelti savo test grafikus

Chart

Description automatically generated

**11 pav**. confusion matricx

# Išvados

* Išvados ir rezultatai;

# Naudotos literatūros sąrašas.

Tips For Using Dropout

The original paper on Dropout provides experimental results on a suite of standard machine learning problems. As a result they provide a number of useful heuristics to consider when using dropout in practice.

Generally, use a small dropout value of 20%-50% of neurons with 20% providing a good starting point. A probability too low has minimal effect and a value too high results in under-learning by the network.

Use a larger network. You are likely to get better performance when dropout is used on a larger network, giving the model more of an opportunity to learn independent representations.

Use dropout on incoming (visible) as well as hidden units. Application of dropout at each layer of the network has shown good results.

Use a large learning rate with decay and a large momentum. Increase your learning rate by a factor of 10 to 100 and use a high momentum value of 0.9 or 0.99.

Constrain the size of network weights. A large learning rate can result in very large network weights. Imposing a constraint on the size of network weights such as max-norm regularization with a size of 4 or 5 has been shown to improve results.

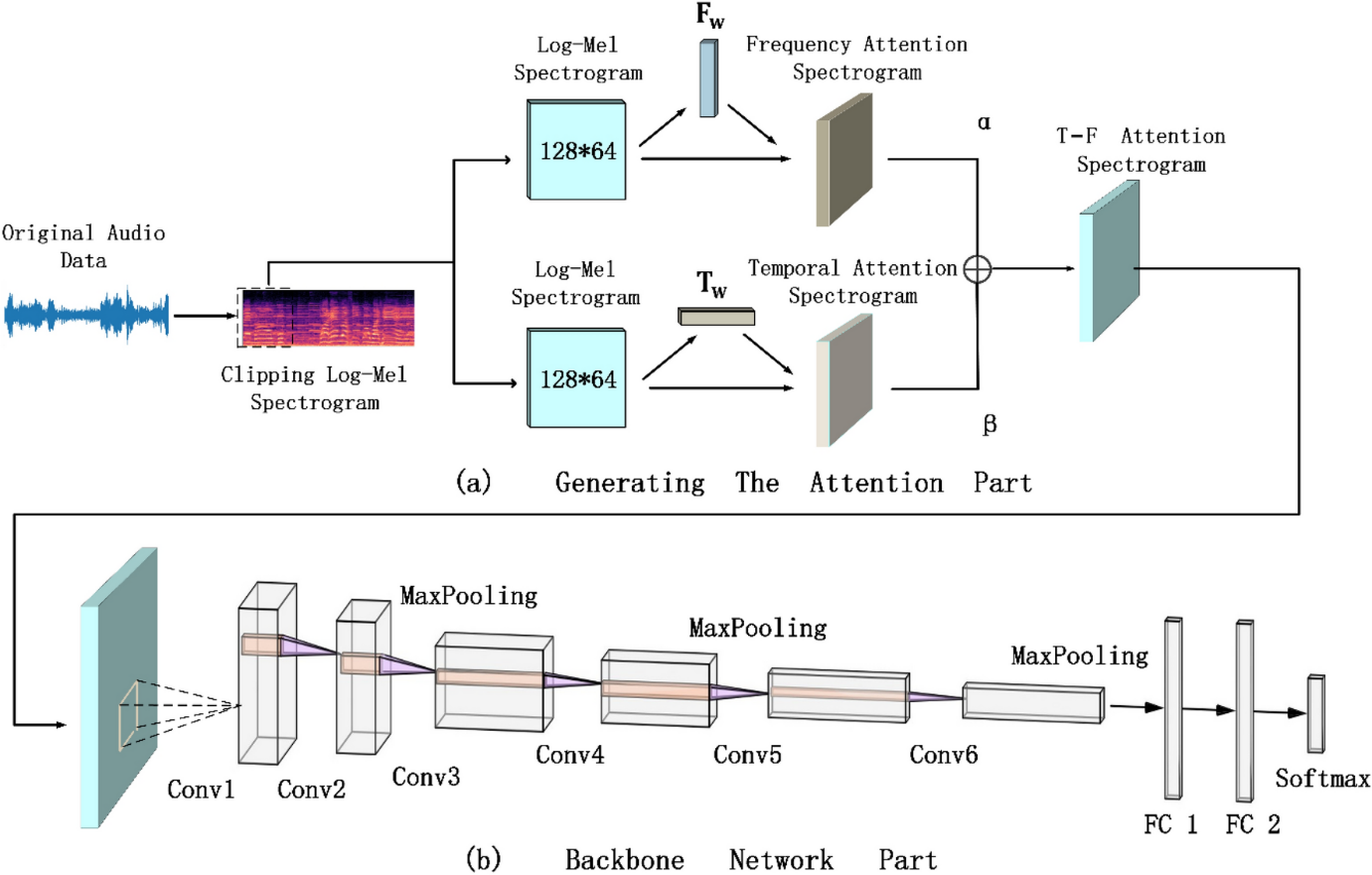
Dropout:

<https://machinelearningmastery.com/dropout-regularization-deep-learning-models-keras/>

???????????

tf.set\_random\_seed(0)

np.random.seed(0)



There are several methods for calculating a correlation value. The most popular one is [*Pearson Correlation Coefficient*](https://en.wikipedia.org/wiki/Pearson_correlation_coefficient). Nevertheless, it should be noticed that it measures only linear relationship between two variables. In other words, it may not be able to reveal a nonlinear relationship. The value of Pearson correlation ranges from -1 to +1, where +/-1 describes a perfect positive/negative correlation and 0 means no correlation.

<https://www.kaggle.com/christianlillelund/classify-mnist-audio-using-spectrograms-keras-cnn>

our audio problem to a image problem.

<https://towardsdatascience.com/audio-deep-learning-made-simple-sound-classification-step-by-step-cebc936bbe5>

<https://stackoverflow.com/questions/56719138/how-can-i-save-a-librosa-spectrogram-plot-as-a-specific-sized-image/57204349#57204349>

Kiti straipsniai

<https://github.com/karolpiczak/ESC-50>

Chart, line chart

Description automatically generatedChart, line chart

Description automatically generated

Batch normalization is a **technique for training very deep neural networks that standardizes the inputs to a layer for each mini-batch**. This has the effect of stabilizing the learning process and dramatically reducing the number of training epochs required to train deep networks

<https://www.youtube.com/watch?v=DtEq44FTPM4>

Batch normalization is not recommended as an alternative to proper data preparation for your model.

[6] - <http://www.justinsalamon.com/uploads/4/3/9/4/4394963/salamon_urbansound_acmmm14.pdf>

Alex net

<https://iq.opengenus.org/architecture-and-use-of-alexnet/>