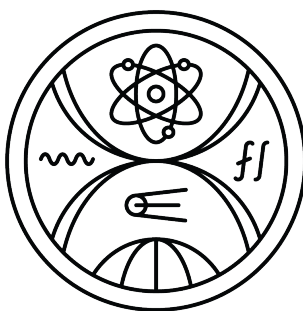


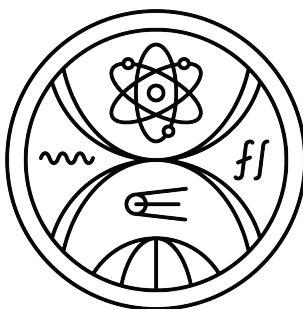
Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky



AUTOMATICKÁ GENERÁCIA ÚROVNÍ PRE RYTMICKÉ POČÍTAČOVÉ HRY

Diplomová práca

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky



AUTOMATICKÁ GENERÁCIA ÚROVNÍ PRE RYTMICKÉ POČÍTAČOVÉ HRY

Diplomová práca

Študijný program: Aplikovaná informatika
Študijný odbor: 18. Informatika
Školiace pracovisko: Katedra aplikovanej informatiky
Školiteľ: Ing. Viktor Kocur, PhD.

Čestne prehlasujem, že túto diplomovú prácu som vypracoval samostatne len s použitím uvedenej literatúry a za pomoci konzultácií u môjho školiteľa.

Bratislava, 2024

.....
Bc. Filip Masný

Pod'akovanie

Abstrakt

Klíčové slová:

Abstract

Keywords:

Obsah

1	Všeobecný prehľad a teória	2
1.1	Rytmické Počítačové Hry a Ich Štruktúra	2
1.2	Rôzne typy rytmických počítačových hier	3
1.3	Hra Beat Saber	3
1.4	Umelá neurónová sieť	4
1.4.1	Trénovanie neurónovej siete	5
1.4.2	Aktivačné funkcie	6
1.5	Optimalizačné algoritmy	7
1.5.1	Stochastický zostup po gradiente	7
1.5.2	Momentum	8
1.5.3	Adam	8
1.6	Konvolučná neurónová sieť	8
1.6.1	Konvolučná vrstva	9
1.6.2	Pooling vrstva	10
1.6.3	Plne prepojená vrstva	11

Zoznam obrázkov

1.1	Ukážka neurónovej siete	5
1.2	Ukážka Forward pass (Modre sipky) a Backpropagation (Cervene sipky)	6
1.3	Príklad operácie 2D konvolúcie. Štvorce so šípkami znázorňujú, ako bol vypočítaný ľavý horný prvok výstupnej matice po aplikácii jadra na príslušnú ľavú hornú oblasť vstupnej matice.	9
1.4	Príklad MAX poolingu (hore), príklad priemerného poolingu (dole) . .	10
1.5	Príklad globalneho poolingu	11

Zoznam tabuliek

Úvod

Kapitola 1

Všeobecný prehľad a teória

Umelá inteligencia (UI) zaznamenala za posledných niekoľko desaťročí obrovský rast a vývoj, ktorý posunul hranice možností počítačov. Na čele UI je oblasť počítačového videnia využívajúca konvolučné neurónové siete, ktoré dosahujú prelomové výsledky v úlohách rozpoznávania obrazov a objektov.

V tejto kapitole sa budeme venovať Rytmickým počítačovým hram a teóriám umelej inteligencie a CNN, preskúmame prelomové architektúry VGG [42] a ResNet. Začneme s Rytmickými počítačovými hrami s ich štruktúrou. Potom položením základov neurónovej siete. A na koniec tréňovaním neurónovej siete, Forward pass, aktivačnými funkciami a známym algoritmom Backpropagation na prispôbenie váh, aby sa neurónová sieť mohla učiť.

1.1 Rytmické Počítačové Hry a Ich Štruktúra

Rytmické počítačové hry predstavujú špecifický žáner videohier, kde je kľúčovým aspektom hrateľnosti hudba a rytmus. Hráči v týchto hrách nie sú len pasívnymi pozorovateľmi, ale aktívne interagujú s herným prostredím, pričom ich akcie sú synchronizované s hudobnými prvkami. V týchto hrách hráči často vykonávajú preddefinované akcie, ako sú stlačenia kláves alebo tlačidiel v rytmickej poradi, v súlade s hudbou a vizuálnymi indikáciami na obrazovke.

Kľúčové charakteristiky rytmickej počítačových hier zahŕňajú:

- **Hudobný Prvok** - Hudba nie je len pozadím, ale aktívne ovplyvňuje hrateľnosť a vizuálny dizajn hry. Hráči reagujú na hudobné prvky, ako sú rytmus a melódia.
- **Vizuálne Indikácie** - Na obrazovke sa zobrazujú grafické indikácie alebo symboly, ktoré signalizujú hráčovi, kedy a aké akcie je potrebné vykonať. Vizuálne prvky sú synchronizované s hudobným pozadím.

- **Preddefinované Akcie** - Hráči vykonávajú preddefinované akcie, ako sú stlačenia kláves alebo tlačidiel, v súlade s hudbou a vizuálnymi indikáciami na obrazovke.
- **Úrovne alebo Levely** - Hry často obsahujú štruktúrované úrovne alebo levely, kde sa obtiažnosť postupne zvyšuje. Hráči dosahujú úspech tým, že dokončujú úrovne s vysokou presnosťou a bodovým skóre.

Tieto hry nie len poskytujú zábavu, ale tiež stimulujú hráčové vnímanie hudby a rytmu, vytvárajúc jedinečný zážitok spojením digitálnej zábavy s hudobným umením.

1.2 Rôzne typy rytmických počítačových hier

Rozlíšenie medzi rôznymi typmi rytmických hier môže byť užitočné na porozumenie rôznych prístupov k tomuto žánru. Tu sú niektoré základné typy rytmických hier a ich charakteristiky:

- **Tanečné hry** - Hráči tancujú na hudbu a snažia sa napodobniť tanečné kroky, ktoré sú zobrazené na obrazovke, poprípade majú tanečnú podložku, na ktorej sa nachádzajú tlačidlá, na ktoré je potrebné stúpiť keď sa na obrazovke. Príkladom tanečných hier sú *Just Dance* alebo *Dance Dance Revolution*.
- **Hry s Nástrojmi** - Hráči hrajú na reálne hudobné nástroje, ako sú gitara alebo klávesy. Hráči musia zahrať správne tóny v správnom čase. Príkladom hier s nástrojmi sú *Guitar Hero* alebo *Rock Band*.
- **Hry s Dotykovým Displejom** - Hráči stláčajú tlačidlá alebo podobné objekty na dotykovom displeji. Príkladom hier s dotykovým displejom sú *Beatstar* alebo *osu!*.
- **Hry vo Virtuálnej Realite** - Hráči interagujú s roznými objektami v hernom prostredí vo virtuálnej realite. Príkladom hry vo virtuálnej realite je *Beat Saber*.

Poslednému zmieňovanému typu a hre *Beat Saber* sa budeme venovať v tejto práci.

1.3 Hra Beat Saber

Beat Saber je hra pre virtuálnu realitu, ktorá bola vyvinutá a vydaná českým herným štúdiom Beat Games. Hra bola v predbežnom prístupe vydaná v máji 2018 a neskôr v máji 2019 bola vydaná ako plná verzia. Hra je dostupná pre platformy PlayStation 4, Windows, Meta Quest a ďalšie. Hráč používa ovládače pre virtuálnu realitu na

ovládanie dvoch svetelných mečov, ktoré sú prednastavené na červenú a modrú farbu pre ľavú a pravú ruku. [3]

V každej pesničke hra hráčovi predkladá postupnosť približujúcich sa blokov súčasne s rytmom hudby, umiestnených v jednej z 12 možných pozícií na mriežke 4x3. Každý blok môže byť označený šípkou, ktorá naznačuje jednu z ôsmich možných smerov, ktorým musí byť blok správne pretnutý. Existujú aj bloky s bodkami namiesto šípiek, ktoré hráči môžu zasiahnuť z akéhokoľvek smeru. Okrem toho sa objavujú aj bomby, ktoré hráč nesmie zasiahnuť, a prekážky v podobe približujúcich sa stien, ktorým sa je potrebné vyhnúť hlavou.

Beat Saber bol dodaný s desiatimi piesňami, ale bol rozšírený niekoľkými balíkmi s obsahom na stiahnutie a aktualizáciami, ktoré zahŕňajú nové piesne. Niekoľko z nich obsahuje originálne skladby, ale niekoľko balíkov obsahuje licencované piesne s hudbou a špeciálnymi úrovňami od rôznych umelcov. Okrem toho komunita vytvorila modifikácie pre Beat Saber, ktoré umožňujú vlastné piesne a mapy.

Počas hry sa objavujú bloky, ktoré majú rôzne špeciálne efekty, ako napríklad blok s vedenou čiarou alebo blok pozostávajúci z viacerých menších blokov. Zároveň má hra veľmi bohato editovateľné vizuálne efekty, ako napríklad laserové show a podobne. V aktuálnej verzii hry sú k dispozícii 5 rôznych herných režimov: Standard, No Arrows (bloky je možné pretnúť v akomkoľvek smere), One Saber (používa sa len jeden meč), 90 Degrees a 360 Degrees (bloky sa približujú v rôznych smeroch).

Hra má tiež niekoľko úrovní obtiažnosti, ktoré sú Easy, Normal, Hard, Expert a Expert+. Tieto úrovne obtiažnosti sa líšia počtom blokov, rýchlosťou približovania sa blokov, počtom prekážok a podobne.

V našej práci sa budeme venovať režimu Standard, ktorý je najpoužívanejší a najviac rozšírený. A zároveň sa budeme venovať trom základným objektom v hre, ktorými sú bloky, prekážky a bomby. Zameriame sa aj na všetky úrovne obtiažnosti, ktoré sú dostupné v hre.

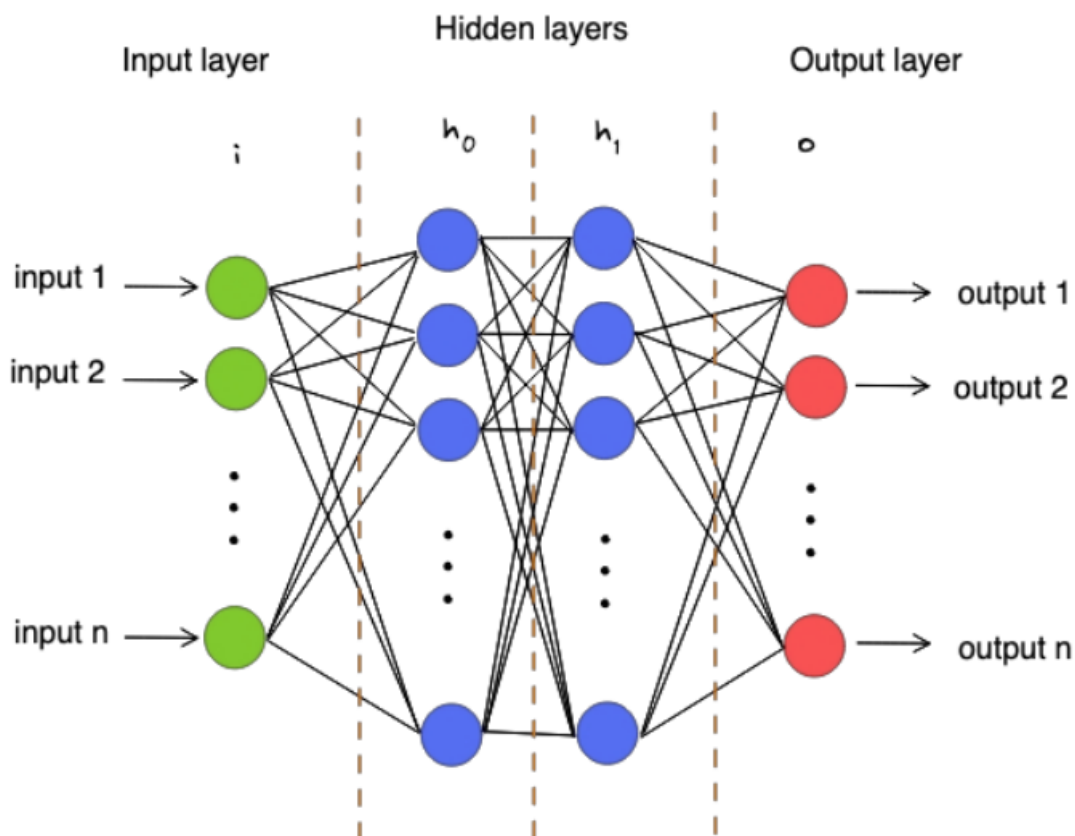
1.4 Umelá neurónová sieť

Koncept umelej neurónovej siete (ANN) je voľne inšpirovaný fungovaním ľudského mozgu. Neuróny v ANN sú podobne prepojené a aktivované ako v ľudskom mozgu. ANN prijíma vstupné údaje a prechádza nimi cez aktivačnú funkciu, v dôsledku čoho sa niektoré neuróny v sieti aktivujú a vytvárajú výstup.

Univerzálna aproximačná veta hovorí, že vždy môžeme nájsť dostatočne veľkú neurónovú sieť so správnou sadou váh, ktorá dokáže správne predpovedať akýkoľvek výstup pri akomkoľvek vstupe.

Neurónová sieť sa skladá zo vstupnej, skrytej a výstupnej vrstvy. Ak sú neuróny

v každej vrstve prepojené s neurónmi v inej vrstve, potom je neurónová sieť plne prepojená neurónová sieť. Hodnoty neurónov v sieti závisia od vstupných údajov a pridanej nelinearity. Spojenia medzi neurónmi sú takzvané váhy.



Obr. 1.1: Ukážka neurónovej siete

1.4.1 Trénovanie neurónovej siete

Forward pass

Forward pass je jednou z rozhodujúcich zložiek procesu tréovania neurónových sietí. Dopredu sa šíriacim spôsobom neurónová sieť vypočítava výstupné predpovede na základe vstupných údajov. Začína sa na vstupnej vrstve, kde sa prijímajú údaje, a pokračuje cez každú ďalšiu vrstvu v sieti. Počas tohto prechodu sa vstupné údaje transformujú prostredníctvom série lineárnych a nelineárnych operácií, ako sú násobenie matíc a aktivačné funkcie. Tieto transformácie prebiehajú v každej vrstve a výstup z jednej vrstvy slúži ako vstup pre ďalšiu vrstvu. Posledná výstupná vrstva generuje predpovede siete.

Pred začatím tréovania siete sa náhodne vygenerujú hodnoty váh. Pre lepšiu konvergenciu váh sa odporúča vzorkovať ich z Heovho rozdelenia.

Stratová funkcia

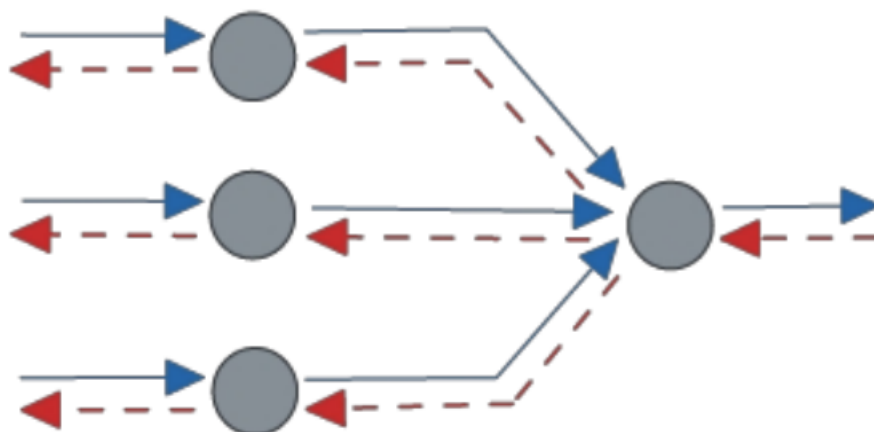
Stratová alebo chybová funkcia udáva, ako model funguje. V procese tréningu neurónových sietí chceme minimalizovať stratovú funkciu úpravou váh siete. Na výpočet stratovej funkcie zvyčajne potrebujeme anotované hodnoty základných pravdivých hodnôt. Pre stratovú funkciu môžeme zvoliť jednoduchú strednú kvadratickú chybu.

Ďalším príkladom stratovej funkcie pre problémy binárnej klasifikácie je binárna krížová entropia

Pri binárnej klasifikácii sú predpovedané kategórie práve 0 a 1.

Backpropagation

Cieľom algoritmu spätného šírenia je vypočítať deriváciu stratovej funkcie vzhľadom na každú váhu v sieti. To sa môže použiť v spojení s optimalizačným algoritmom založeným na gradiente na minimalizáciu stratovej funkcie. Algoritmus začína od výstupnej vrstvy a prechádza do vstupnej vrstvy. Aby sme vypočítali, ako veľmi prispieva každá váha siete k celkovej chybe, musíme vypočítať gradienty pomocou reťazového pravidla.



Obr. 1.2: Ukážka Forward pass (Modre sipky) a Backpropagation (Cervene sipky)

1.4.2 Aktivačné funkcie

Aktivačná funkcia je funkcia, ktorá sa aplikuje na výstup každého neurónu v sieti. Aktivačná funkcia pridáva do modelu nelinearitu, takže sieť môže aproximovať takmer akúkoľvek funkciu. Ak sa aktivačná funkcia do siete nezapracuje, model bude môcť vykonávať len lineárnu klasifikáciu

Sigmoid

Sigmoid je aktivačná funkcia, ktorá je definovaná ako $\sigma(x) = \frac{1}{1+e^{-x}}$. Táto funkcia je spojitá a diferencovateľná, čo je dôležité pre algoritmus. Táto aktivačná funkcia sa

používala v začiatkoch. Aktivačná funkcia však stláča výstup na rozsah medzi $[0, 1]$. Výsledkom je sigmoida výstup funkcie sa nasýti pri vysokých a nízkych vstupných hodnotách, čo má za následok problém miznúceho gradientu. Problém miznúceho gradientu sa vzťahuje na situáciu, keď sa gradient účelovej funkcie týkajúci sa parametra blíži k nule, čo spôsobuje malé aktualizácie parametrov počas tréningu siete pomocou metódy stochastického zostupu po gradiente. To má za následok nedostatočné zmeny parametrov siete, a sieť sa tak nedokáže správne učiť. Okrem toho neprítomnosť nulového výstupu prispieva k suboptimálnej konvergencii.

ReLU

ReLU je aktivačná funkcia, ktorá je definovaná ako $f(x) = \max(0, x)$. Táto funkcia je spojitá, ale nie je diferencovateľná v bode $x = 0$. Táto funkcia je v súčasnosti najpoužívanejšia aktivačná funkcia. ReLU je jednoduchá na výpočet a má výhodu, že sa nenasýti pri vysokých vstupných hodnotách. ReLU však môže mať problém s mŕtvymi neurónmi. Mŕtvy neurón je neurón, ktorý má všetky váhy nastavené na záporné hodnoty. V tomto prípade sa výstup neurónu vždy rovná 0 a gradient je vždy 0. To znamená, že neurón sa už nikdy nebude učiť, pretože sa nikdy neaktualizujú jeho váhy. Tento problém sa dá vyriešiť použitím variantu ReLU, ktorý sa nazýva Leaky ReLU. Leaky ReLU je definovaná ako $f(x) = \max(0.01x, x)$.

Softmax

Softmax je aktivačná funkcia, ktorá je definovaná ako $\sigma(x)_i = \frac{e^{x_i}}{\sum_{j=1}^K e^{x_j}}$. Táto funkcia je spojitá a diferencovateľná. Táto funkcia sa používa na klasifikáciu viacerých tried. Výstup tejto funkcie je vždy v rozsahu $[0, 1]$ a súčet výstupov je vždy 1. Táto funkcia sa používa na klasifikáciu viacerých tried. Výstup tejto funkcie je vždy v rozsahu $[0, 1]$ a súčet výstupov je vždy 1. Táto funkcia sa používa na klasifikáciu viacerých tried. Výstup tejto funkcie je vždy v rozsahu $[0, 1]$ a súčet výstupov je vždy 1.

1.5 Optimalizačné algoritmy

Optimalizačné algoritmy sa používajú na minimalizáciu stratovej funkcie. V tejto sekcii sa budeme venovať niektorým základným optimalizačným algoritmom.

1.5.1 Stochastický zostup po gradiente

Stochastický zostup po gradiente (SGD) je najzákladnejší optimalizačný algoritmus. SGD je iteratívny algoritmus, ktorý sa používa na minimalizáciu stratovej funkcie. Algoritmus začína náhodným výberom počiatočných váh. Potom sa vypočíta gradient

stratovej funkcie vzhľadom na každú váhu v sieti. Váhy sa aktualizujú v smere gradientu stratovej funkcie. Tento proces sa opakuje, kým sa váhy nemenia alebo kým sa nedosiahne maximálny počet iterácií.

1.5.2 Momentum

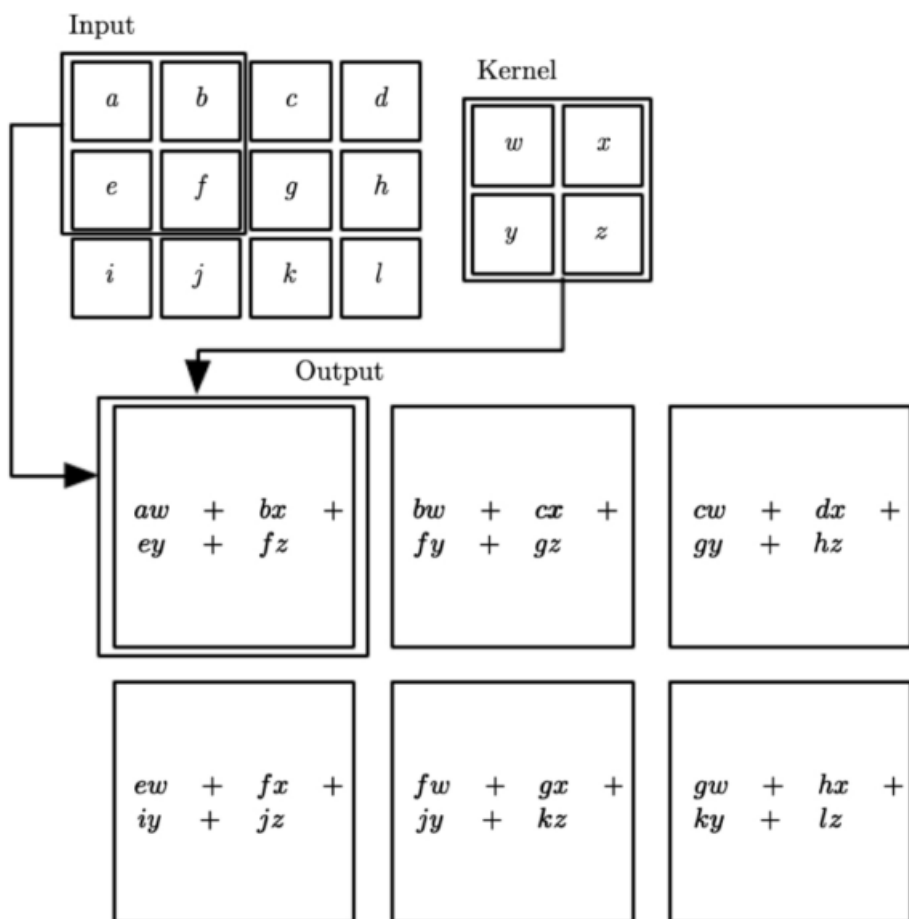
Momentum je optimalizačný algoritmus, ktorý sa používa na minimalizáciu stratovej funkcie. Tento algoritmus je podobný SGD, ale má výhodu, že sa môže vyhnúť lokálnym minimám. Tento algoritmus používa koncept momentum, ktorý je definovaný ako exponenciálny pohyblivý priemer gradientov.

1.5.3 Adam

Adam je optimalizačný algoritmus, ktorý sa používa na minimalizáciu stratovej funkcie. Tento algoritmus je kombináciou algoritmov RMSProp a Momentum. Tento algoritmus používa koncept exponenciálneho pohyblivého priemeru gradientov a exponenciálneho pohyblivého priemeru druhých momentov gradientov.

1.6 Konvolučná neurónová sieť

Konvolučná neurónová sieť (CNN) je špeciálna viacvrstvová neurónová sieť pre priestorové údaje. Tradičná CNN sa skladá z jednej alebo viacerých konvulčných vrstiev, vrstiev združovania, po ktorých nasleduje jedna alebo viacero plne prepojených vrstiev a na konci je výstupná vrstva.



Obr. 1.3: Príklad operácie 2D konvolúcie. Štvorce so šípkami znázorňujú, ako bol vypočítaný ľavý horný prvok výstupnej matice po aplikácii jadra na príslušnú ľavú hornú oblasť vstupnej matice.

1.6.1 Konvolučná vrstva

Konvolučná vrstva je základným stavebným kameňom CNN. Pozostáva z konvolučného jadra (filtrov). Jadro sa konvolúciou spája so vstupnými údajmi, čo vedie k extrakcii príznakov uložených v mape príznakov. Jadro možno opísať ako maticu s naučenými hodnotami, ktoré sú známe ako váha jadra. Príklad 2D jadra na obrázku 1.5. V procese tréningu sa váhy jadra CNN aktualizujú, čím sa zo vstupu extrahujú ďalšie podstatné vlastnosti. Na extrakciu príznakov zo vstupného obrazu sa vykonáva operácia konvolúcie. Operácia sa začína v ľavom hornom rohu vstupnej matice. Jadro je posunuté horizontálne aj vertikálne. Po každom posune sa vykoná bodový súčin vynásobením a sčítaním príslušných hodnôt v jadre a vstupnom obraze a vytvorí sa jedna skalárna hodnota na výstupe mapy príznakov.

Konvolučná vrstva má ďalšie dva parametre. Prvým je krok, ktorý definuje krok posunu jadra, a druhým je výplň. Výplňanie rieši problém rýchlo stráca informácie o prvkoch na okrajoch obrázkov. Padding je pridávanie núl okolo obrazu, zväčšovanie

vstupnej veľkosti a pridávanie ďalších informácií o prvkoch na hraniciach do výslednej mapy prvkov.

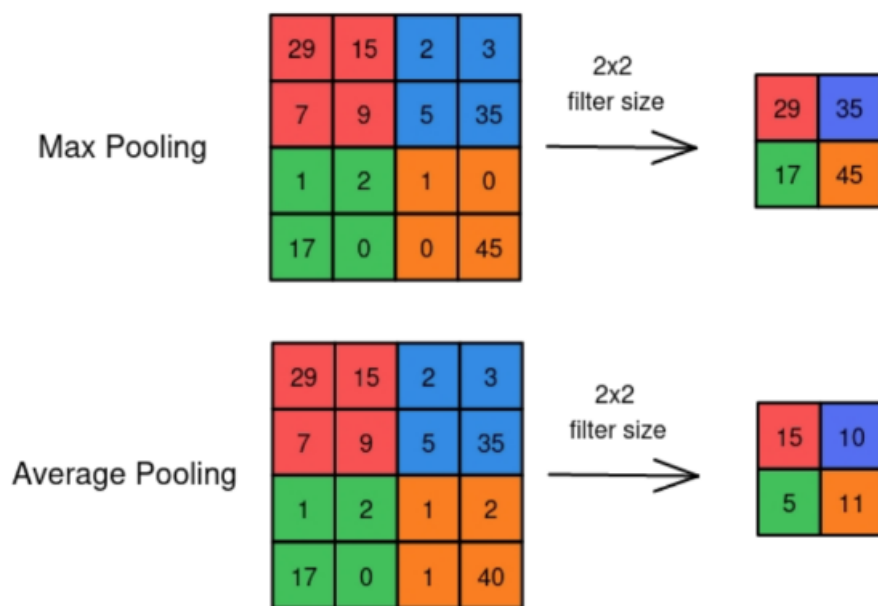
Použitím operácie konvolúcie na vstupné údaje (obrázok, audio) sa zmenia rozmery vstupných údajov. Táto zmena závisí od veľkosti jadra, kroku a výplne (padding).

1.6.2 Pooling vrstva

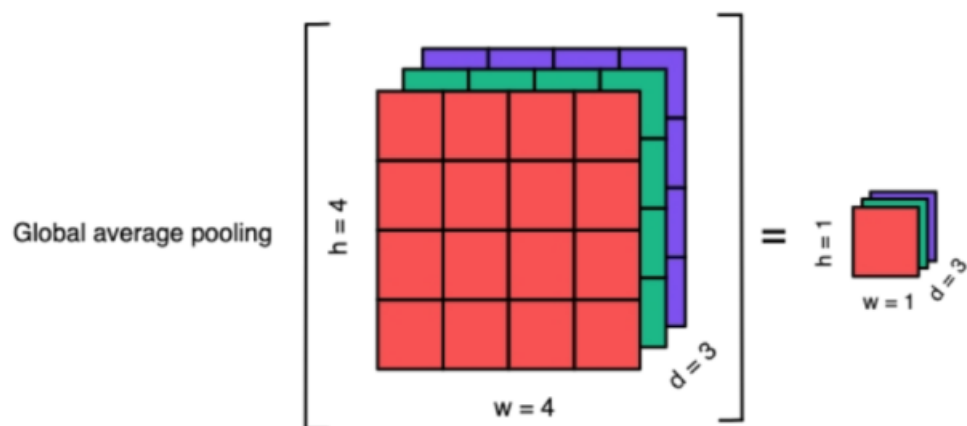
Pooling vrstvy sa bežne vkladajú medzi konvolučné vrstvy. Pooling vrstva teda sumarizuje prvky v oblasti mapy prvkov vytvorenej predchádzajúcou konvolúciou vrstvy. Nasledujúce výpočty sa teda vykonávajú na zosumarizovaných prvkoch namiesto presne umiestnených prvkov vytvorených konvolučnou vrstvou. Tento proces zabezpečuje väčšiu odolnosť modelu voči zmenám polohy prvku vo vstupnom obraze. Keďže združovacia vrstva sumarizuje prvky, znižuje priestorové rozmery. Existujú tri hlavné operácie pooling: priemer, maximum a globálne združovanie. Priemerný pooling berie priemernú hodnotu z pooling filtra.

Max pooling má rovnaký postup, ale berie maximálnu hodnotu.

Globálny priemerný pooling sa používa na redukcii trojrozmerných máp prvkov na jednorozmerné vektory.



Obr. 1.4: Príklad MAX pooling (hore), príklad priemerného pooling (dole)



Obr. 1.5: Príklad globalneho poolingu

1.6.3 Plne prepojená vrstva

Posledným stupňom konvolučnej neurónovej siete je plne prepojená vrstva. V tejto vrstve sú neuróny navzájom prepojené. Táto vrstva sa používa napríklad na klasifikáciu objektov na vstupnom obraze na základe extrahovaných funkcií z predchádzajúcich vrstiev, v našom prípade konvolučných vrstiev.

Záver

Literatúra

- [1] Daniel PW Ellis. Beat tracking by dynamic programming. *Journal of New Music Research*, 36(1):51—60, 2007.
- [2] Bertrand David Miguel Alonso and Gaël Richard. Tempo and beat estimation of musical signals. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR), Barcelona, Spain*, 2004.
- [3] Nicholas Sutrich and Cale Hunt. Beat saber: Everything you need to know about the vr rhythm game, 2020.