

**UNIVERZITA PARDUBICE**  
Fakulta elektrotechniky a informatiky

**ANALÝZA POHYBOVÝCH DAT PACIENTŮ PO OPERACI  
ROVNOVÁŽNÉHO ÚSTROJÍ**

Bc. Jan Holan

Diplomová práce  
2024

Univerzita Pardubice  
Fakulta elektrotechniky a informatiky  
Akademický rok: 2023/2024

# ZADÁNÍ DIPLOMOVÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Bc. Jan Holan**  
Osobní číslo: **I22180**  
Studijní program: **N0714A150005 Automatické řízení**  
Téma práce: **Analýza pohybových dat pacientů po operaci rovnovážného ústrojí**  
Zadávající katedra: **Katedra řízení procesů**

## Zásady pro vypracování

Práce je zaměřena na pokročilé metody analýzy biomedicínských dat a jejich rekonstrukci. Pacienti po operaci rovnovážného ústrojí mají dočasné problémy s rovnováhou a ovládním mímických svalů. Práce bude zaměřena na analýzu dat (která jsou k dispozici se svolením Kliniky ORL FNKV Praha) a na objektivní evaluaci rehabilitačního procesu takového pacienta (především zmeny v pohyblivosti).

Cílem práce bude:

- 1) vypracovat literární rešerší tématu,
- 2) nastudovat vybrané metody 3D rekonstrukce a analýzy datových záznamů
- 3) aplikovat tuto metodiku na reálná data

Rozsah pracovní zprávy:  
Rozsah grafických prací:  
Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam doporučené literatury:

Shayestegan, M.; Kohout, J.; Štícha, K.; Mareš, J. Advanced Analysis of 3D Kinect Data: Supervised Classification of Facial Nerve Function via Parallel Convolutional Neural Networks. *Appl. Sci.* 2022, *12*, 5902. <https://doi.org/10.3390/app12125902>

Červená, L.; Kříž, P.; Kohout, J.; Vejvar, M.; Verešpejová, L.; Štícha, K.; Crha, J.; Tmková, K.; Chovanec, M.; Mareš, J. Advanced Statistical Analysis of 3D Kinect Data: A Comparison of the Classification Methods. *Appl. Sci.* 2021, *11*, 4572. <https://doi.org/10.3390/app11104572>

Vedoucí diplomové práce: **doc. Ing. Jan Mareš, PhD.**  
Katedra řízení procesů

Datum zadání diplomové práce: **8. listopadu 2023**  
Termín odevzdání diplomové práce: **17. května 2024**

## **Prohlášení**

Prohlašuji:

Práci s názvem Analýza pohybových dat pacientů po operaci rovnovážného ústrojí jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Beru na vědomí, že v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, a směrnicí Univerzity Pardubice č. 7/2019 Pravidla pro odevzdávání, zveřejňování a formální úpravu závěrečných prací, ve znění pozdějších dodatků, bude práce zveřejněna prostřednictvím Digitální knihovny Univerzity Pardubice.

V Pardubicích dne

Jan Holan v.r.

## **Poděkování**

Chtěl bych poděkovat za věcné rady a připomínky v průběhu psaní vedoucímu práce doc. Ing. Janu Marešovi, Ph.D. Poděkování patří také kolegům z výzkumného týmu doc. Mareše, konkrétně Ing. Janu Kohoutovi, Ph.D. a Ing. Karlu Štíchovi. V poslední řadě bych chtěl poděkovat za podporu přátelům a rodině, kteří mě v průběhu psaní práce podporovali.

V Pardubicích dne

Jan Holan v.r.

## **ANOTACE**

*Práce je zaměřena na pokročilé metody analýzy biomedicínských dat a jejich rekonstrukci. Pacienti po operaci rovnovážného ústrojí mají dočasné problémy s rovnováhou a ovládním mimických svalů. Práce bude zaměřena na analýzu dat (která jsou k dispozici se svolením kliniky ORL FNKV Praha) a objektivní evaluaci rehabilitačního procesu takového pacienta (především změny v pohyblivosti).*

## **KLÍČOVÁ SLOVA**

*Analýza biomedicínských dat, klasifikace, neuronové sítě, operace, pohybová data, rehabilitace, rekonstrukce dat, rovnovážné ústrojí*

## **TITLE**

*ANALYSIS OF MOVEMENT DATA OF PATIENTS AFTER BALANCE SYSTEM SURGERY*

## **ANNOTATION**

*The work is focused on advanced methods of biomedical data analysis and reconstruction. Patients after balance system surgery have temporary problems with balance and control of facial muscles. The work will be focused on data analysis (available with permission of the ENT clinic of FNKV Prague) and objective evaluation of rehabilitation process of such patient (especially changes in mobility).*

## **KEYWORDS**

*Biomedical data analysis, classification, neural networks, surgery, motion data, rehabilitation, data reconstruction, balance system*

## OBSAH

	SEZNAM ZKRATEK A ZNAČEK .....	9
	SEZNAM OBRÁZKŮ .....	10
	SEZNAM TABULEK .....	13
	ÚVOD.....	14
1	ANATOMIE SLUCHOVĚ ROVNOVÁŽNÉHO ÚSTROJÍ.....	16
1.1	Vestibulární ústrojí .....	16
1.2	Ústrojí zrakové .....	17
1.3	Anatomie a funkce hlavových nervů .....	17
1.4	Onemocnění lícního nervu.....	18
1.5	Příznaky parézy lícního nervu .....	20
1.6	Léčba parézy lícního nervu.....	20
1.7	House-brackmannova stupnice .....	21
2	DATOVÁ ANALÝZA .....	23
2.1	Dataset .....	23
2.2	3D rekonstrukce.....	23
2.2.1	Proces 3D rekonstrukce .....	24
3	KLASIFIKACE POMOCÍ STATISTICKÝCH METOD .....	25
3.1	Klasifikace dat .....	25
3.2	Regresí metody .....	25
3.3	Logistická regrese.....	26
4	KLASIFIKACE POMOCÍ UMĚLÉ INTELIGENCE .....	27
4.1	Algoritmus k-nejbližších sousedů .....	27
4.2	Rozhodovací stromy .....	27
4.3	Náhodné lesy .....	28
5	UMĚLÉ NEURONOVÉ SÍTĚ .....	30
5.1	Základní terminologie.....	30
5.2	Proces učení .....	33
5.2.1	Hebbův zákon učení .....	33
5.2.2	Učení s učitelem .....	34
5.2.3	Učení bez učitele .....	34
5.3	Konvoluční neuronové sítě.....	34
5.3.1	Architektura CNN.....	34

5.4	Rekurentní neuronové sítě .....	35
5.4.1	Architektura RNN.....	36
5.5	Sítě s dlouhou krátkodobou pamětí .....	36
5.5.1	Architektura LSTM .....	37
5.5.2	Buňka LSTM .....	38
5.6	Uzavřená rekurentní jednotka.....	39
5.6.1	Architektura GRU.....	39
6	SBĚR DAT .....	41
6.1	Vizualizace ve 3D.....	42
6.2	Přístup k datovému souboru .....	43
7	KLASIFIKACE POMOCÍ LSTM.....	46
7.1	Předzpracování dat .....	46
7.2	Definice a Kompilace modelu .....	47
7.3	Trénování Modelu .....	48
8	KLASIFIKACE POMOCÍ GRU .....	54
8.1	Definice a kompilace modelu .....	54
8.2	Trénování modelu.....	54
9	KLASIFIKACE POMOCÍ CNN .....	59
9.1	Předzpracování dat .....	59
9.2	Definice a kompilace modelu .....	59
9.3	Trénování modelu.....	60
10	DISKUZE .....	65
	ZÁVĚR.....	68
	POUŽITÁ LITERATURA .....	70

## SEZNAM ZKRATEK A ZNAČEK

2D	dvourozměrný
3D	trojrozměrný
ANN	umělé neuronové síť
CNN	konvoluční neuronové síť
CT	centrální tomografie
DT	rozhodovací stromy
EMG	elektromyografie
FNKV	Fakultní nemocnice Královské Vinohrady
GRU	uzavřená opakující se jednotka
kNN	k-nejbližších sousedů
LSTM	sítě s dlouhou krátkodobou pamětí
MLP	vícevrstvé perceptrony
MR	magnetická rezonance
OOB	vzorky mimo soubor
RF	náhodné lesy
RNN	rekurentní neuronové síť
VPN	virtuální privátní síť

## SEZNAM OBRÁZKŮ

Obr. 1.1 - Anatomie lidského ucha, převzato z (Národní zdravotnický informační portál, neuvedeno) .....	16
Obr. 1.2 - Lící nerv, převzato z (Lynch, 2006) .....	18
Obr. 4.1 - Rozhodovací stromy, převzato z (Illustrating the random forest algorithm in TikZ, 2019).....	28
Obr. 4.2 - Náhodné lesy, převzato z (Illustrating the random forest algorithm in TikZ, 2019).....	29
Obr. 5.1 - Dopředná vícevrstvá neuronová síť .....	30
Obr. 5.2 - Architektura umělého neuronu, převzato z (Ansh, 2021) .....	31
Obr. 5.3 - Aktivační funkce Softmax.....	32
Obr. 5.4 - Aktivační funkce Tanh .....	32
Obr. 5.5 - Aktivační funkce lineárně saturovaná .....	32
Obr. 5.6 - Aktivační funkce ReLU .....	32
Obr. 5.7 - Sigmoidní aktivační funkce.....	32
Obr. 5.8 - Lineární aktivační funkce.....	32
Obr. 5.9 - Skoková aktivační funkce .....	33
Obr. 5.10 - Architektura sítě CNN, převzato z (Li, 2022).....	35
Obr. 5.11 - Architektura RNN, převzato z (Olah, 2015) .....	36
Obr. 5.12 - Architektura buňky LSTM, převzato z (Elmaz, 2021).....	38
Obr. 5.13 - Architektura buňky GRU, převzato z (Nosouhain, 2021).....	40
Obr. 6.1 - Vykreslení obličejových značek.....	41
Obr. 6.2 - 3D rekonstrukce obličeje pomocí MediaPipe .....	42
Obr. 6.3 - Kód pro načtení a vizualizaci dat z databáze .....	43
Obr. 6.4 - Prostředí aplikace SQL Server Management Studio pro připojení k databázi.....	44
Obr. 6.5 - Kód pro připojení k databázi .....	44
Obr. 6.6 - Pokračování kódu pro připojení k databázi.....	45
Obr. 7.1 – LSTM: Předzpracování dat.....	46
Obr. 7.2 - LSTM: Načtení a zpracování dat .....	47
Obr. 7.3 - LSTM: Definice modelu .....	48
Obr. 7.4 - LSTM: Trénování modelu.....	48
Obr. 7.5 - LSTM: Validace pro experiment č. 1 .....	49
Obr. 7.6 - LSTM: Ztráta pro experiment č. 1 .....	49

Obr. 7.7 - LSTM: Definice modelu pro experiment č. 2 .....	49
Obr. 7.8 - LSTM: Validace pro experiment č. 2.....	50
Obr. 7.9 - LSTM: Ztráta pro experiment č. 2 .....	50
Obr. 7.10 - LSTM: Validace pro experiment č. 3.....	50
Obr. 7.11 - LSTM: Ztráta pro experiment č. 3 .....	50
Obr. 7.12 - LSTM: Trénování modelu pro experiment č. 4.....	51
Obr. 7.13 - LSTM: Validace modelu pro experiment č. 4.....	51
Obr. 7.14 - LSTM: Ztráta modelu pro experiment č. 4 .....	51
Obr. 7.15 - LSTM: Validace modelu pro experiment č. 5.....	52
Obr. 7.16 - LSTM: Ztráta modelu pro experiment č. 5 .....	52
Obr. 7.17 - LSTM: Validace modelu pro experiment č. 6.....	52
Obr. 7.18 - LSTM: Ztráta modelu pro experiment č. 6 .....	52
Obr. 7.19 - LSTM: Validace modelu pro experiment č. 7.....	53
Obr. 7.20 - LSTM: Ztráta modelu pro experiment č. 7 .....	53
Obr. 7.21 - LSTM: Validace modelu pro experiment č. 8.....	53
Obr. 7.22 - LSTM: Ztráta modelu pro experiment č. 8 .....	53
Obr. 8.1 - GRU: Definice modelu.....	54
Obr. 8.2 - GRU: Validace modelu pro experiment č. 1 .....	55
Obr. 8.3 - GRU: Ztráta modelu pro experiment č. 1.....	55
Obr. 8.4 - GRU: Validace modelu pro experiment č. 2.....	55
Obr. 8.5 - GRU: Ztráta modelu pro experiment č. 2.....	55
Obr. 8.6 - GRU: Validace modelu pro experiment č. 3 .....	56
Obr. 8.7 - GRU: Ztráta modelu pro experiment č. 3.....	56
Obr. 8.8 - GRU: Validace modelu pro experiment č. 4.....	56
Obr. 8.9 - GRU: Ztráta modelu pro experiment č. 4.....	56
Obr. 8.10 - GRU: Validace modelu pro experiment č. 5 .....	57
Obr. 8.11 - GRU: Ztráta modelu pro experiment č. 5.....	57
Obr. 8.12 - GRU: Validace modelu pro experiment č. 6.....	57
Obr. 8.13 - GRU: Ztráta modelu pro experiment č. 6.....	57
Obr. 8.14 - GRU: Validace modelu pro experiment č. 7 .....	58
Obr. 8.15 - GRU: Ztráta modelu pro experiment č. 7.....	58
Obr. 8.16 - GRU: Validace modelu pro experiment č. 8 .....	58
Obr. 8.17 - GRU: Ztráta modelu pro experiment č. 8.....	58
Obr. 9.1 - CNN: Předzpracování dat.....	59

Obr. 9.2 - CNN: Definice modelu.....	60
Obr. 9.3 - CNN: Validace modelu pro experiment č. 1 .....	60
Obr. 9.4 - CNN: Ztráta modelu pro experiment č. 1 .....	60
Obr. 9.5 - CNN: Validace modelu pro experiment č. 2 .....	61
Obr. 9.6 - CNN: Ztráta modelu pro experiment č. 2.....	61
Obr. 9.7 - CNN: Validace modelu pro experiment č. 4 .....	62
Obr. 9.8 - CNN: Ztráta modelu pro experiment č. 4.....	62
Obr. 9.9 - CNN: Validace modelu pro experiment č. 3 .....	62
Obr. 9.10 - CNN: Ztráta modelu pro experiment č. 3.....	62
Obr. 9.11 - CNN: Validace modelu pro experiment č. 5 .....	63
Obr. 9.12 - CNN: Ztráta modelu pro experiment č. 5.....	63
Obr. 9.13 - CNN: Validace modelu pro experiment č. 6 .....	63
Obr. 9.14 - CNN: Ztráta modelu pro experiment č. 6.....	63
Obr. 9.15 - CNN: Validace modelu pro experiment č. 7 .....	64
Obr. 9.16 - CNN: Ztráta modelu pro experiment č. 7.....	64
Obr. 9.17 - CNN: Validace modelu pro experiment č. 8 .....	64
Obr. 9.18 - CNN: Ztráta modelu pro experiment č. 8.....	64

## **SEZNAM TABULEK**

Tab. 1.1 - Tabulka hlavových nervů .....	17
Tab. 1.2 - House-Brackmannova stupnice, převzato z (Alicandri-Ciufelli, 2013).....	22
Tab. 6.1 - Přehled rehabilitačních cvičení .....	41
Tab. 10.1 - Tabulka výsledků trénování pro LSTM .....	65
Tab. 10.2 - Tabulka výsledků trénování pro GRU .....	65
Tab. 10.3 - Tabulka výsledků trénování pro CNN .....	66
Tab. 10.4 - Tabulka výsledků pro všechny sítě .....	66

# ÚVOD

V oblasti biomedicínského výzkumu hrají klíčovou roli pokročilé metody analýzy a rekonstrukce dat. Tyto metody umožňují vědcům extrahovat cenné informace z komplexních datových sad, které by jinak mohly zůstat skryty. V této diplomové práci se zaměřím na aplikaci těchto metod v kontextu rehabilitace pacientů nejen po operaci rovnovážného ústrojí.

Pacienti po operaci rovnovážného ústrojí často trpí dočasnými problémy s rovnováhou a ovládním mimických svalů. Tyto potíže mohou mít značný dopad na jejich kvalitu života a každodenní aktivity. Rehabilitace hraje klíčovou roli v procesu zotavování a pomáhá pacientům dosáhnout optimálního zotavení.

V první části provedu rozsáhlou literární rešerši daného tématu. Prostuduji existující metody 3D rekonstrukce a analýzy datových záznamů a vyberu ty, které jsou nejvhodnější pro účely této práce. V druhé části této práce aplikuji vybrané metody na reálná data pacientů, která jsou k dispozici se svolením Kliniky ORL Fakultní nemocnice Královské Vinohrady Praha. Pro analýzu dat budu využívat klasifikaci pomocí konvolučních a rekurentních neuronových sítí.

## Cíle práce

Analýza pohybových dat pacientů po operaci rovnovážného ústrojí za pomoci neuronových sítí:

- Hlavním cílem této práce je analyzovat pohybová data pacientů po operaci rovnovážného ústrojí s cílem identifikovat vzorce a změny v jejich pohybu během rehabilitačního procesu;
- K dosažení tohoto cíle budou použity neuronové sítě, které jsou výkonným nástrojem pro analýzu komplexních dat a rozpoznávání vzorců;
- Očekává se, že výsledky analýzy pomohou lépe porozumět procesu zotavování pacientů a efektivněji plánovat rehabilitační programy.

Vytvoření 3D modelu pacienta po provedení operačního zákroku, až po ukončení rehabilitace:

- Druhým cílem práce je vytvořit 3D model pacienta před a po operaci rovnovážného ústrojí a během rehabilitačního procesu;

- Tento 3D model bude sloužit k vizualizaci změn pohybu pacienta a k lepšímu pochopení dopadu operace a rehabilitace na jeho tělo;
- Očekává se, že 3D model bude užitečným nástrojem pro lékaře, fyzioterapeuty a pacienty samé.

Objektivizace dat a zaměření se na problémy lícního nervu pacientů v dané oblasti:

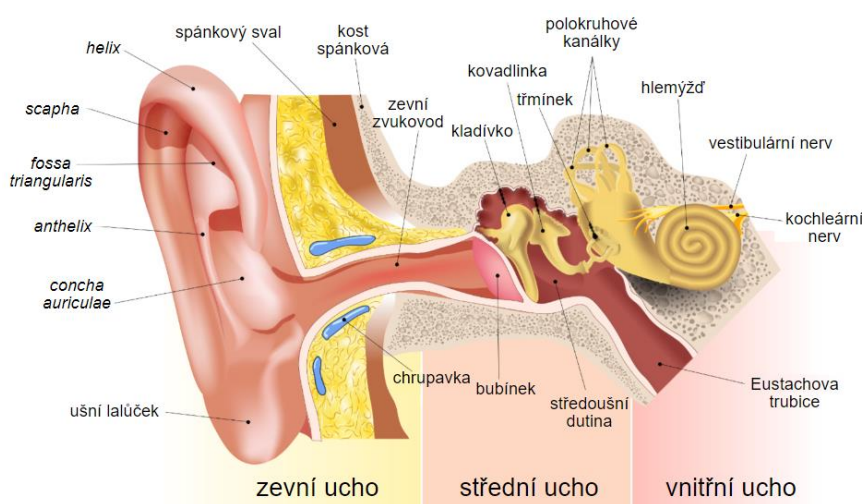
- Třetím cílem práce je objektivizovat data o pohybu pacientů a zaměřit se na problémy lícního nervu, které se u pacientů po operaci lícního nervu mohou vyskytovat;
- Očekává se, že by výzkum této práce mohl vést k vývoji nových diagnostických a léčebných nástrojů pro pacienty s poruchami lícního nervu.

# 1 ANATOMIE SLUCHOVĚ ROVNOVÁŽNÉHO ÚSTROJÍ

Rovnováha je v lidském těle zajišťována několika smyslovými systémy. Nejzásadnějším ústrojím je vestibulární aparát, který se nachází v zadní části vnitřního ucha a je klíčový pro náš smysl pro rovnováhu a pohyb. Další důležité součásti jsou smyslová ústrojí sluchové a zrakové. Řídicím centrem rovnovážného ústrojí je mozeček.

Jakékoli změny v tomto systému mohou způsobit příznaky, jako jsou závratě, rozmazané vidění, nerovnováha a závrať, což jsou indikátory vertiginózního syndromu. Závrať je uváděna jako jeden z nejčastějších příznaků na světě.

Sluchově – rovnovážné ústrojí je uloženo oboustranně ve spánkové kosti v lebce – nejtvrdější kosti lidského těla (Fife, 2010).



Obr. 1.1 - Anatomie lidského ucha, převzato z (Národní zdravotnický informační portál, neuvedeno)

## 1.1 VESTIBULÁRNÍ ÚSTROJÍ

Toto ústrojí je součástí vnitřního ucha, dělíme jej na dvě části: statické a kinetické čidlo. K vnímání polohy těla je určeno statické čidlo, které je uloženo v blanitém labyrintu a informace z něj jsou převáděny do mozku. K vnímání pohybu těla je pak určeno kinetické čidlo. To se nachází v amplitudách polokruhovitých kanálků a slouží k ovlivnění rotací hlavy a také ovlivňuje vzpřímený postoj člověka (Santos, 2017).

## 1.2 ÚÚSTROJÍ ZRAKOVÉ

Oči se podílejí na řízení rovnováhy. Zrakové ústrojí je citlivé na světelné vlny, což nám umožňuje vnímat barvy, velikost a vzdálenost objektů, a hlavně umožňuje orientaci v prostoru. Orgánem zrakového ústrojí jsou pak oční koule a další přídatné orgány (Prasad, 2011).

## 1.3 ANATOMIE A FUNKCE HLAVOVÝCH NERVŮ

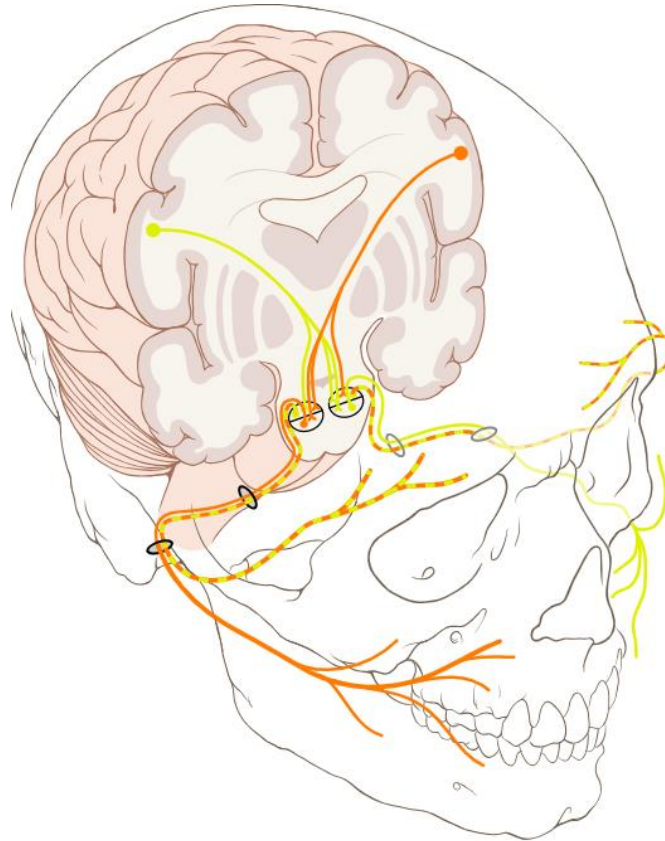
Hlavové nervy (nervi craniales) jsou, společně s nervy míšními, součástí periferního nervového systému a jsou tvořeny 12 páry nervů vystupujících z mozkového kmene (výjimkou je I. a II. hlavový nerv). Lebku opouštějí prostřednictvím otvorů v bázi lebky a označujeme je římskými číslicemi:

Tab. 1.1 - Tabulka hlavových nervů, převzato z (Amber, 2023)

Přehled hlavových nervů		
I.	olfactorius	čichový nerv
II.	opticus	zrakový nerv
III	oculomotorius	okohybný nerv
IV	trochlearis	kladkový nerv
V	trigeminus	trojklanný nerv
VI	abducens	odtahovací nerv
VII	facialis	lícní nerv
VIII	vestibulocochlearis	rovnovážný a sluchový nerv
IX	glossopharyngeus	jazykohltanový nerv
X	vagus	bloudivý nerv
XI	accessorius	přídavný nerv
XII	hypoglossus	podjazykový nerv

Na rozdíl od míšních nervů není u některých hlavových nervů při výstupu z centrálního nervového systému rozlišen přední a zadní kořen. Inervační oblasti hlavových nervů jsou hlava a krk, kromě toho nervus vagus inervuje břišní a hrudní orgány. Mezi okohybné hlavové nervy patří okohybný nerv (I.), kladkový nerv (II.) a odtahovací nerv. (VI.), smyslovými nervy jsou čichový nerv (I.), zrakový nerv (II.) a rovnovážný a sluchový nerv (VIII.) Optický nerv (II.) je výběžkem mezimozku (diencephala), svědčí o tom uspořádání jeho obalů. Z boku mozkového kmene odstupují nervy postranního smíšeného systému jazykohltanový nerv (IX.), bloudivého nervu (X.), přídavného nervu (XI.). XII. hlavový nerv – podjazykový nerv za vývoje vzniká splnutím kraniálních míšních nervů inervujících okcipitální myotomy. O jeho vývojovém vztahu ke krčním nervům svědčí i spojení XII. s ansa cervicalis prima z C1 a C2. Vlastní motorické buňky jsou uloženy v jádrech hlavových nervů v mozkovém kmeni. Tam vede z kůry tractus corticonuclearis, který se kříží. Vyjma část jádra nervus

facialis, který inervuje dolní části obličeje, má většina hlavových nervů oboustrannoukorovou reprezentaci. Proto se jednostranná léze hlavového nervu ve většině případech neprojeví (Ambler, 2023; Druga, 2011).



Obr. 1.2 - Lící nerv, převzato z (Lynch, 2006)

## 1.4 ONEMOCNĚNÍ LÍCNÍHO NERVU

Lící nerv je sedmý lebeční nerv a nese několik typů nervových vláken, kterými ovládá pohyby a výraz obličeje. Skládá se z motorické, parasymptické a senzorické větve. Vychází z mozkového kmene prostřednictvím 3 různých jader, přičemž motorické jádro je největší. Po opuštění mozkového kmene lze lící nerv rozdělit na 5 intrakraniálních segmentů (cisternální, kanálový, labyrintový, tympanický a mastoidní) a intraparotický extrakraniální segment. Lící nerv také přenáší nervy, které se podílejí na chuti předních 2/3 jazyka a produkci slz (slzná žláza). Má malé větve, které se podílejí na mírnění naší citlivosti na hlasitosti hluku a několik dalších svalů, které se nepodílejí na běžné mimice obličeje (Ottaiano, 2023).

Buňky, které přenášejí informace v rámci lícího nervu, pocházejí ze základny mozku neboli mozkového kmene v oblasti nazývané jako pons. Aby se nerv dostal do obličeje, musí

projít základnou lebky. Na této cestě se lícni nerv pohybuje v těsné blízkosti osmého lebečního nervu, vestibulokochleárního nervu, který je zodpovědný za sluch a rovnováhu (Wong, 2016).

Patofyziologie onemocnění lícniho nervu:

- **Idiopatická** – Bellova obrna se vyskytuje v 50 % případů – léčí se kortikosteroidy;
- **Traumata** – iatrogenní, operace v oblasti spánkové kosti, úrazy parotis, zlomeniny spánkové kosti;
- **Infekce a záněty** – především neuroinfekce a neurotropní viry (herpes zoster oticus, varicela-zoster (pásový opar), Lymská borelióza, záněty parotis, záněty v oblasti spánkové kosti – středoušní zánět, mastoidita);
- **Nádory** – spánková kost, parafarynx. Vyšetřovací metody se zaměřením na spánkovou kost nativní RTG snímek, Computerová tomografie, magnetická rezonance, elektromyografie;
- **Paréza lícniho nervu (paresis n. facialis)** – obrna lícniho nervu, porucha funkce, resp. též struktury lícniho nervu (n. facialis) s následnou obrnou obličejového svalstva a poruchou vzhledu a pohybů obličeje. Má buď centrální, nebo periferní charakter.

Typy postižení lícniho nervu:

- a) **Centrální paréza** vzniká při postižení supranukleárním (od jádra lícniho nervu výše), např. přicévní mozkové příhodě;
- b) **Periferní paréza** vzniká např. při lymeské borrelióze, virových infekcích, onemocněních ucha, nádorech glandula parotis aj. Někdy příčina není známá (Bellova paréza). P. bývá obv. jednostranná, např. při borrelióze však může být i oboustranná. Ve stomatologii je nejdůležitější déle trávající periferní (chabá) paréza jako následek nesprávně prováděné lokální anestezie (mandibulární anestezie na foramen mandibulae);
- c) **Paréza esenciální** je typ periferní léze vznikající náhle v důsledku výrazného prochladnutí. Nazývá se též revmatická obrna nebo obrna „e frigore“;
- d) **Nukleární paréza** je typ periferní léze vznikající lézí jádra tohoto nervu (malacie, hemoragie, nádor, zánět). Současně bývá poškozena i kortikospinální dráha (kontralaterální hemiplegie –hemiplegia alternans media) a často i jádro VI. (konvergentní strabismus);
- e) **Sekundární paréza** je typ léze nervi facialis způsobené patologickým ložiskem různé etiologie – otogenní obrna VII. (způsobená uzurací stěny canalis facialis rozsáhlým

zánětem středoušní dutiny), neuritida (neurosyfilis, polyradikuloneuritida), neurinom akustiku, tumory, arachnoiditida pontocerebelárního úhlu, traumata lebeční spodiny aj. V popředí periferní p. je asymetrie obličeje (pokles ústního koutku, nestejná šíře oční štěrbin), bývá pokles spodního víčka s následnou neschopností úplně zavřít oko (nebezpečí vzniku lagofthalmu), neschopnost svraštit čelo, zapískat aj. Bývají poruchy citlivosti příslušné oblasti, hyperakuzie (porucha am. stapedius), porucha vnímání chuti v předních dvou třetinách jazyka, porucha sekrece slz a slin. Pro centrální p. je charakteristické postižení dolní větve se šetřením inervace m. frontalis am. orbicularis oculi (typický pokles ústního koutku po cévní mozkové příhodě (Vokurka, 2023).

## 1.5 PŘÍZNAKY PARÉZY LÍCNÍHO NERVU

- 1) Ochrnutí mimických svalů – nejvíce viditelné a nejvíce psychicky traumatizující pro pacienta – vyhlazené vrásky na postižené straně obličeje, neschopnost cenit zuby, zapískat, nemožnost zavřít oční štěrbinu, pocit ztuhlosti obličeje;
- 2) Hyperakuze – zvýšená citlivost na zvuky, lze odstranit příčinu hluku nebo ucpávkou zevního zvukovodu;
- 3) Suchost oka vede k pocitu pálení v oku a hrozí poškození rohovky odíráním (asi ve 30 %), zpravidla je i zánět spojivek, suchost oka i zánět se stupňují;
- 4) Změny kožního cití za boltcem často bolest v malém okrsku;
- 5) Chuť zpravidla není změněna z důvodu zajištění druhou stranou nervu a dvěma dalšími nervy. Obtěžující se stává v okamžiku změny cití, vnímáním „kovové chuti“, ale to je v případě regrese (zlepšení). Což je prognosticky příznivým znamením.

Vyšetřovací metody u parézy lícního nervu:

- Nativní RTG snímek;
- výpočetní tomografie (CT);
- magnetická rezonance (MR);
- elektromyografie (EMG) (Kusyn, 2020).

## 1.6 LÉČBA PARÉZY LÍCNÍHO NERVU

Rehabilitace ochrnutého obličeje se zabývá jak fyzickými, tak psychologickými aspekty postižení obličeje. Pacienti s obrnou obličeje trpí nedostatkem mimiky a řadou funkčních omezení, včetně neschopnosti mluvit, artikulačních obtíží a zrakových poruch.

Pokud je navíc porušena schopnost vyjadřovat emoce, jsou pacienti klasifikováni jako negativně afektovaní. Kvalita života je často narušena a může se objevit úzkost a deprese. Nedostatečná léčba následků spojených s neúplnou nebo aberantní regenerací nervu může vést k dysfunkčním pohybům obličeje a devastujícím psychosociálním následkům. Obnovení funkce obličeje na nejvyšší možnou míru vede ke zlepšení sebevědomí a kvality života.

Aby se funkce lícního nervu, a tedy i pohyblivost obličejových svalů po operaci obnovila jsou zapotřebí pravidelné kontroly u lékaře. Hodnocení lékaře je klíčové pro úspěšné zvládnutí rehabilitace. Hodnocení lékaře má význam:

- a) motivuje pacienta;
- b) plánuje se další postup rehabilitace;
- c) slouží jako ukazatel úspěšnosti rehabilitace. (Robinson, 2018)

## **1.7 HOUSE-BRACKMANNOVA STUPNICE**

V dnešní době existuje několik způsobů klinického hodnocení stupně funkce obličeje. Klasifikace obličejové paralýzy je nezbytná pro určení a hlášení spontánního průběhu obličejové paralýzy a zejména výsledků lékařské nebo chirurgické léčby. Nejčastějším prostředkem pro vyhodnocení funkce lícního nervu jsou klinické stupnice. V průběhu let bylo navrženo několik měřících stupnic, nicméně i přes svá značná omezení, tak nejpoužívanější stupnicí je House-Brackmannova klasifikační stupnice. Tento systém používá šestibodovou stupnici, kde stupeň I odpovídá normálnímu stavu a stupeň VI úplné paralýze obličeje (Alicandri-Ciuffelli, 2013).

Tab. 1.2 - House-Brackmannova stupnice, převzato z (Alicandri-Ciuffelli, 2013)

Stupeň postižení	Funkce obličejových svalů
I normální	Normální funkce obličeje ve všech oblastech.
II mírná disfunkce	Mírná slabina patrná pouze při bližším zkoumání. V klidu: normální symetrie čela, schopnost zavřít oko s mírným úsilím, mírná asymetrie, schopnost pohybovat koutky úst s maximálním úsilím a mírnou asymetrií.
III střední disfunkce	Zřejmý, ale ne znetvořující rozdíl mezi dvěma stranami, ne funkční porucha. V klidu: normální symetrie a tonus. Pohyb: mírný až žádný pohyb čela, schopnost pohybovat koutky úst s maximálním úsilím a zřejmou nesymetrií.
IV středně závažná disfunkce	Zjevná slabost a/nebo znetvořující asymetrie. V klidu: normální symetrie a tonus. Pohyb: žádný pohyb čela, neschopnost úplně zavřít oko při maximálním úsilí.
V těžká disfunkce	Jen sotva znatelný pohyb. V klidu: možná asymetrie s poklesem ústního koutku. Pohyb: žádný pohyb čela, neúplné zavření oka a jen nepatrný pohyb víčka při maximálním úsilí, nepatrný pohyb ústního koutku.
VI úplná paralýza	Žádný pohyb

## 2 DATOVÁ ANALÝZA

Data jsou základem průmyslového i akademického výzkumu. Při provádění kontrolovaných pozorování a experimentů odborníky se získávají vysoce kvalitní výzkumná data. V dřívějších dobách se data sbírala pomocí ručních záznamů, zatímco s rozvojem moderních technologií přibývá dat, která jsou zaznamenávána pomocí senzorů kontinuálně během časového intervalu nebo přerušovaně v několika oddělených časových bodech. Taková data poté můžeme vyhodnocovat pomocí statistických metod. Statistika je vědní obor, který se zabývá sběrem, uspořádáním a analýzou dat a vyvozováním závěrů ze vzorků pro celou danou množinu (Wang, 2016)

### 2.1 DATASET

Dataset je strukturovaný soubor dat uspořádaných a uložených společně za účelem analýzy nebo zpracování. Všechna data v datasetu spolu obvykle nějakým způsobem souvisí a pochází z jednoho zdroje nebo jsou určena pro jeden projekt. Datový soubor může obsahovat mnoho různých typů dat, od číselných hodnot po text, obrázky nebo zvukové záznamy. K datům lze obvykle přistupovat jednotlivě, v kombinaci nebo je spravovat jako celek. Vlastnosti, které by měl mít kvalitní dataset:

- Reprezentativnost – Dataset věrně odráží zkoumaný jev či proces;
- Přesnost – Data jsou bez chyb a duplicit;
- Úplnost – Dataset obsahuje všechny relevantní informace pro daný účel;
- Konzistence – Data jsou formátována a strukturována jednotně (Ding, 2014).

### 2.2 3D REKONSTRUKCE

Trojrozměrná (3D) rekonstrukce je proces generování 3D reprezentace objektu nebo scény ze sady dvourozměrných (2D) snímků nebo videí. Jedná se o složitý úkol, který zahrnuje extrakci geometrických informací, jako je: hloubka, tvar a textura ze vstupních dat. Konečným cílem je znovu vytvořit digitální model, který přesně odráží strukturu a vzhled reálného objektu. Oblast 3D rekonstrukce se vzájemně prolíná s počítačovým viděním, počítačovou grafikou a zpracováním obrazu. Mezi techniky používané při 3D rekonstrukci patří detekce a shoda prvků, kalibrace kamery, stereo vidění, struktura na základě pohybu a další. Algoritmy umělé inteligence a strojového učení hrají zásadní roli při automatizaci a zdokonalování těchto technik, čímž se proces zrychluje a zpřesňuje (Yuniarti, 2019).

## 2.2.1 Proces 3D rekonstrukce

Rekonstrukce dat se skládá z několika kroků:

- 1) **Sběr dat** – V první fázi se shromažďují vstupní data, ať už pořízením fotografií z různých úhlů, natočením videa nebo naskenováním objektu. Nejrozšířenější a nejoblíbenější jsou optické metody rekonstrukce modelu, které mají nejrychlejší sběr dat. K tomuto účelu využívají princip triangulace, strukturovaného osvětlení a zpracování obrazu. Poloha bodů je získána pomocí polohy a úhlu mezi světelnými zdroji a fotosenzory v triangulační technice. Vzory odražených paprsků se používají k získání bodových dat ve strukturovaných světelných datech. V technice zpracování obrazu se k určení souřadnic a výšky používají stereofonní dvojice snímků;
- 2) **Předzpracování dat** – Následuje fáze předzpracování dat, která zahrnuje čištění a úpravu vstupních dat. To může zahrnovat odstranění šumu, korekci zkreslení objektivu a vyrovnání snímků. Kvalitní předzpracování dat je nezbytné pro dosažení přesných výsledků v následujících fázích;
- 3) **Vlastní rekonstrukce dat** – V jádru procesu 3D rekonstrukce leží algoritmy, které extrahují trojrozměrné informace z 2D dat. Mezi běžné techniky patří stereofotografie, strukturované světelné skenování a fotogrammetrie. Každá z nich má své silné a slabé stránky a volba techniky závisí na specifických vlastnostech dat a požadované přesnosti;
- 4) **Vizualizace a analýza** – Výsledný 3D model se poté vizualizuje pomocí počítačové grafiky, čímž se umožňuje detailní analýza a interpretace dat. Vizualizace může zahrnovat různé pohledy na model, měření vzdáleností a objemů, simulaci fyzikálních jevů a další;
- 5) **Validace a optimalizace** – Pro zajištění přesnosti a spolehlivosti 3D modelu je nezbytná validace. To zahrnuje porovnání modelu s existujícími daty, jako jsou měření objektu nebo reference z jiných zdrojů. Na základě validace se model může dále optimalizovat pro dosažení co nejpřesnějšího zobrazení reality (Intwala, 2016).

## 3 KLASIFIKACE POMOCÍ STATISTICKÝCH METOD

### 3.1 KLASIFIKACE DAT

Klasifikace má dva různé významy. Můžeme mít k dispozici soubor pozorování s cílem zjistit, zda v datech existují třídy nebo shluky. Anebo můžeme s jistotou vědět, že existuje mnoho tříd, a cílem je stanovit pravidlo, podle kterého můžeme nové pozorování zařadit do jedné z existujících tříd. První typ se nazývá učení bez učitele (z angl. Unsupervised Learning), druhým typem je učení s učitelem (z angl. Supervised Learning). Pokud používáme termín klasifikace, hovoříme o učení s kontrolou. Ve statistické literatuře se Supervised Learning obvykle, ale ne vždy, označuje jako diskriminace, čímž se rozumí stanovení klasifikačního pravidla z daných správně klasifikovaných dat (Hand, 1992; Michie, 1995).

Diagnóza nemoci je stanovována pomocí klasifikace. Cílem je identifikovat na základě charakteristických znaků nemocného a nemoc, kterou trpí a případně i stupeň, kterým je stádium nemoci charakterizováno. Rozhodnutí o diagnóze bude na základě zkušeností získaných ze studií jiných pacientů. Dříve byly tyto zkušenosti shromažďovány a porovnávány subjektivně, ale v posledních letech se začaly používat metody objektivnější statistické. V případě této práce je to pacient trpící parézou lícního nervu, charakteristický je pokles koutku úst, pokles očního víčka na stejné straně obličeje, obtížné mluvení, polykání potravy, tekutin. Tyto charakteristické znaky jsou zahrnuty v House-Brackmannově stupnici (Ali, 2016).

### 3.2 REGRESÍ METODY

Lineární regresní model je základem statistiky již přes 30 let a je stále jedním z nejdůležitějších nástrojů. Odhaduje vztah mezi proměnnou odezvy (z angl. response variable) a jednou nebo více explorativními proměnnými (z angl. explanatory variable) pomocí lineární funkce. Explorativní proměnná definuje nezávislou proměnnou nebo prediktivní proměnnou a zahrnuje vstupní hodnotu  $X$ . Proměnná odezvy, často nazývaná závislá proměnná definuje výstupní hodnotu  $Y$ .

Lineární regresní model předpokládá, že regresní funkce  $E(Y|X)$  je lineární na vstupech  $X_1, \dots, X_p$ . Pro předpovídaný výstup s reálnou hodnotou  $Y$  má lineární model tvar:

$$f = \beta_0 + \sum_{i=1}^n X_i \beta_i \quad (3.1)$$

kde  $f$  je regresní (odhadovaná) funkce a  $\beta_i$  jsou regresní parametry. Jednoduchý lineární model můžeme zapsat jako:

$$Y = \beta X + \varepsilon \quad (3.2)$$

kde  $Y$  je matice výsledků,  $X$  matice plánu a  $\varepsilon$  vektor reziduí (náhodných složek).  
Předpoklady pro jednoduchý regresní model:

- je lineární v parametrech,
- regresní parametry  $\beta_i$  mohou nabývat libovolných hodnot,
- normalita reziduí  $\varepsilon$ ,
- nulová střední hodnota reziduí  $\varepsilon$  (Hastie, 2009; Janurová, 2011).

### 3.3 LOGISTICKÁ REGRESE

Jedná se o matematickou statistickou metodu, jejímž cílem je popis a odhad vztahu mezi závislou proměnnou – výstupem a jednou nebo více nezávislými proměnnými, označovanými jako vstupy. Logistická regrese má mnoho podobností s lineární regresí, ale je složitější a hůře se graficky vyhodnocuje. V lékařské literatuře se hojně používá k analýze binárních výsledků dat. Funkce se dá pochopit jako hledání parametrů  $\beta$ :

Model logistické regrese je ve tvaru:

$$p(x) = \frac{1}{1 + \exp(-\beta_0 + \beta_1 x)} \quad (3.3)$$

kde  $p$  je pravděpodobnost v rozsahu mezi 0 a 1, že daný jev nastane  $x$ -krát a  $\beta$  jsou koeficienty modelu.

Zavedme **logit funkci**, která slouží k transformaci pravděpodobnosti na reálné číselné hodnoty. Logit je definován jako:

$$\text{logit}(p(x)) = \ln\left(\frac{p(x)}{1-p(x)}\right) \quad (3.4)$$

Pro logistickou regresi můžeme rovnici přepsat pomocí logitu jako:

$$\text{logit}(p(y = 1)) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k \quad (3.5)$$

kde  $y$  reprezentuje výstup s binomickým rozdělením a lineárními parametry  $\beta$  (Zabor, 2022).

## 4 KLASIFIKACE POMOCÍ UMĚLÉ INTELIGENCE

Klasifikace pomocí umělé inteligence je proces, při kterém algoritmy strojového učení a umělé inteligence přiřazují objekty nebo data do různých tříd nebo kategorií na základě jejich charakteristik nebo vlastností (Lim, 2016).

### 4.1 ALGORITMUS K-NEJBLIŽŠÍCH SOUSEDŮ

Klasifikace založená na algoritmu k-nejbližšího souseda (kNN, z angl. k-nearest neighbors) používá ke klasifikaci přímo data, bez předchozího sestavení modelu. Proto není třeba brát v úvahu žádné podrobnosti konstrukce modelu a jediným nastavitelným parametrem modelu je  $k$ , tedy počet sousedů, které je třeba zahrnout do odhadu příslušnosti ke třídě: hodnota  $P(y|x)$  se vypočítá jednoduše jako poměr členů třídy  $y$  mezi nejbližšími sousedy  $x$ . Změnou  $k$  lze model učinit více nebo méně flexibilním (malé, resp. velké hodnoty  $k$ ). Standardně funkce kNN používá euklidovskou vzdálenost, kterou lze vypočítat pomocí následující rovnice:

$$\begin{aligned} p(y, x) &= \sqrt{(y_1 - x_1)^2 + (y_2 - x_2)^2 + \dots + (y_n - x_n)^2} = \\ &= \sqrt{\sum_{i=1}^n (y_i - x_i)^2} \end{aligned} \quad (4.1)$$

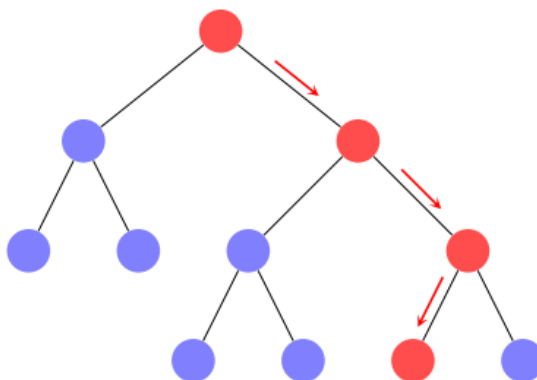
Existují i další metody výpočtu vzdálenosti, jako například Manhattanská vzdálenost (Lim, 2016).

### 4.2 ROZHODOVACÍ STROMY

Rozhodovací stromy (DT, z angl. Decision Trees) jsou metoda strojového učení s učitelem, která operuje bez parametrů. Jejich struktura obsahuje: kořenový uzel, větve a listy (uzly). Kořenový uzel je rodičem všech uzlů a jedná se tak o nejvyšší uzel ve stromu. DT jsou postupným modelem, který efektivně a soudržně spojuje řadu základních testů, kde se v každém testu porovnává číselný znak s prahovou hodnotou. Při tvorbě stromu se využívá např.:

$$S(x_T) = - \sum_{i=1}^n p(x_i) \log_2 p(x_i) \quad (4.2)$$

kde  $S$  je hodnota entropie, jejíž hodnota leží mezi 0 a 1,  $x_T$  jsou trénovací data,  $x_i$  je vstup,  $n$  je počet vstupů a  $p_i(x_i)$  je pravděpodobnost výskytu  $x_i$  v  $x_T$ .



Obr. 4.1 - Rozhodovací stromy, převzato z (Illustrating the random forest algorithm in TikZ, 2019)

Existují různé algoritmy pro zavádění DT shora dolů, např.: ID3 (Iterative Dichotomies 3), nástupce ID3 (C4.5) a CART (Classification and Regression Tree). Některé se skládají z dvou koncepčních fází: růstu a ořezávání (C4.5 a CART). Další algoritmy provádějí pouze funkci růstu (Charbuty, 2021; Patel, 2018)

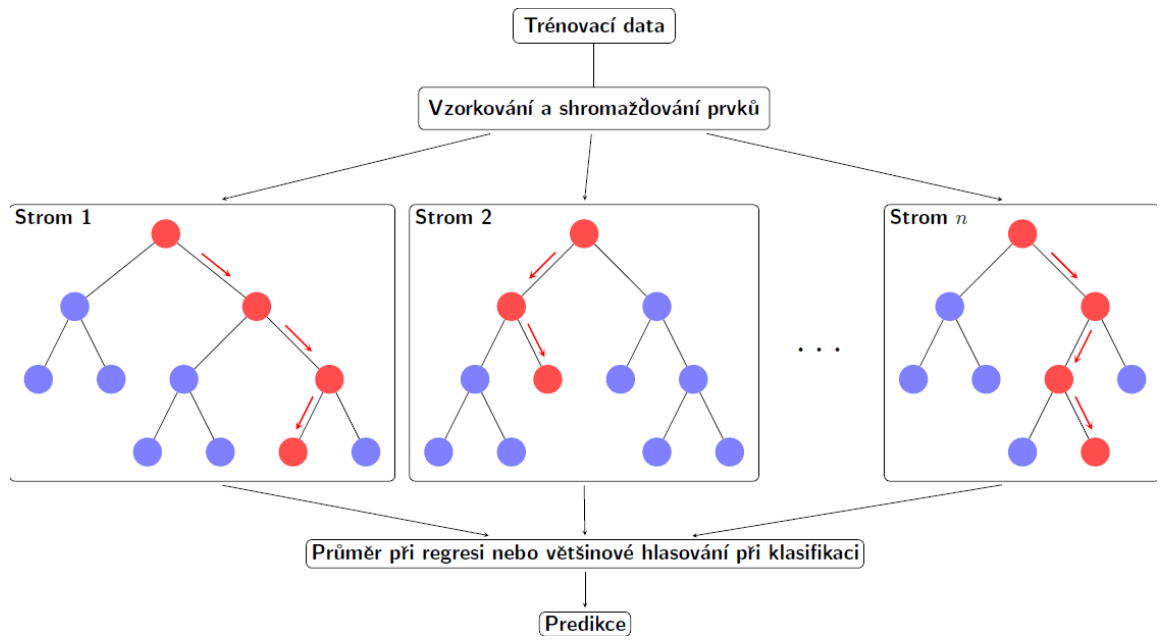
### 4.3 NÁHODNÉ LESY

Náhodné lesy (RF, z angl. Random Forests), které vyvinul Breiman (1984) je soubor CART vycvičených na souborech dat stejné velikosti jako trénovací množina tzv. bootstrapech, vytvořených náhodným převzorkováním na samotné trénovací množině. Nejprve je soubor dat vybrán s náhradou vzorku (bootstrap sampling). Proces agregace nového vzorku tímto způsobem se nazývá „bootstrapová agregace“ nebo „bagging“. Pro průběžné značky lze vypočítat průměr předpovědí stromu jako konečnou hodnotu předpovědi:

$$\hat{l}(x_u) = \frac{1}{T} \sum_{i=1}^T \hat{l}_i(x_u) \quad (4.3)$$

kde  $x_u$  je vstup trénovací množiny,  $T$  je celkový počet stromů v lese a  $\hat{l}_i$  je jeden z množiny rozhodovacích stromů.

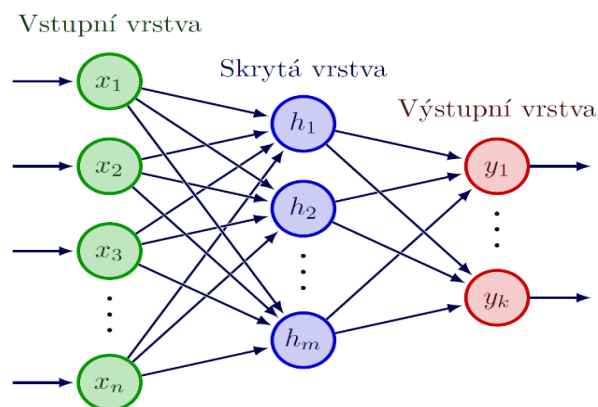
RF typicky fungují tak, že se zkonstruuje mnoho DT, po sestavení stromu se jako testovací soubor použije sada bootstrapů, která neobsahuje žádný konkrétní záznam z původního souboru dat (vzorky mimo soubor (OOB, z angl. out-of bag)). Chybovost klasifikace všech testovacích množin představuje odhad generalizační chyby OOB (Sarica, 2017; Konukoglu, 2020; Rigatti, 2017)



Obr. 4.2 - Náhodné lesy, převzato z (Illustrating the random forest algorithm in TikZ, 2019)

## 5 UMĚLÉ NEURONOVÉ SÍTĚ

Umělé neuronové sítě (ANN, z angl. Artificial Neural Networks) jsou inspirovány funkcí lidského neuronového systému. ANN získávají své znalosti zjišťováním vzorů a vztahů v datech, a učí se (nebo jsou trénovány) na základě zkušeností, nikoliv programováním. Síť je složena z jednotek (neuronů), kde každá má libovolný počet vstupů (synapsí), jeden výstup a transformační (přenosovou funkci), která upravuje signál na vstupu. Jednotlivé neurony jsou propojeny koeficienty (váhami), které tvoří výslednou neuronovou strukturu a jsou uspořádány do vrstev (Ramesh, 2004).

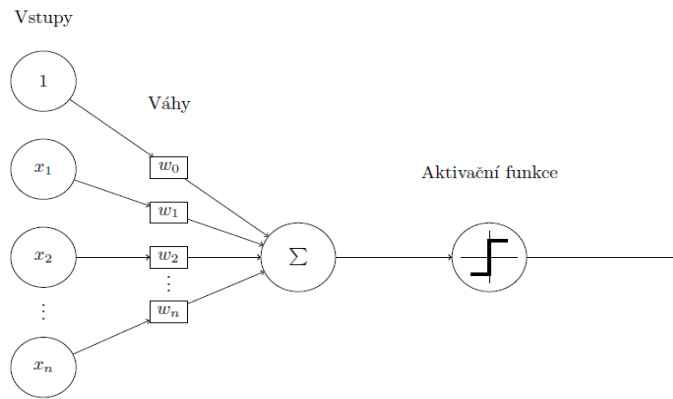


Obr. 5.1 - Dopředná vícevrstvá neuronová síť, převzato z (Ramesh, 2004)

### 5.1 ZÁKLADNÍ TERMINOLOGIE

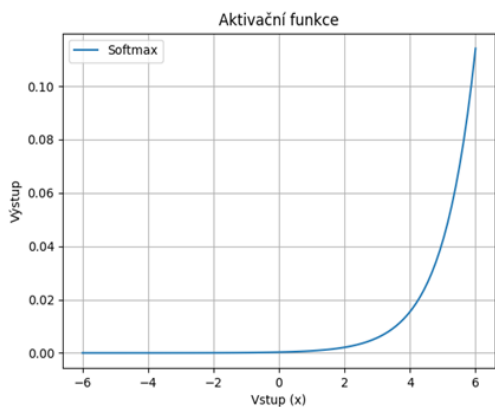
Základní prvky neuronových sítí:

- **neuron (umělý neuron)** – základní stavební blok neuronové sítě, modeluje základní funkčnost biologických neuronů;

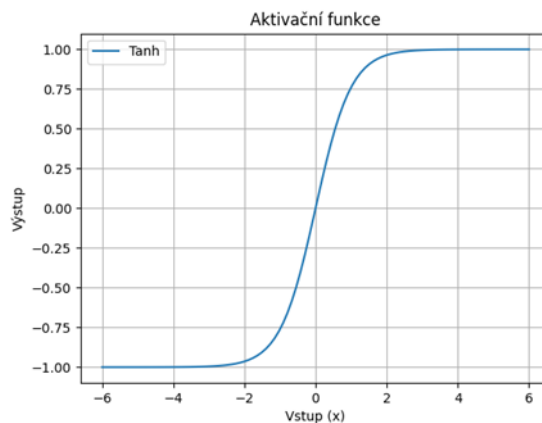


Obr. 5.2 - Architektura umělého neuronu, převzato z  
(Ansh, 2021)

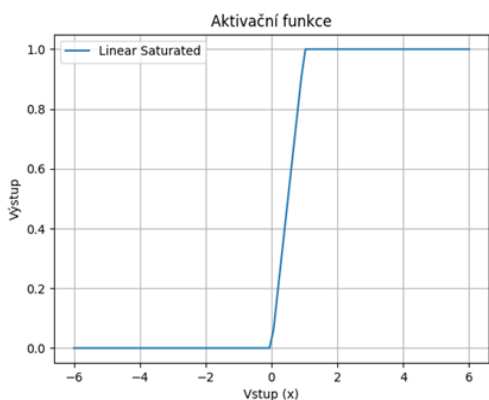
- **vrstva (angl. layer)** – neuronové sítě jsou organizovány do vrstev, každá vrstva může obsahovat různý počet neuronů a můžeme je dělit na:
  - vstupní – první vrstva v síti, která přijímá externí vstupy;
  - skrytá – tato se vrstva se nachází mezi vstupní a výstupní vrstvou;
  - výstupní – poslední vrstva sítě, generuje výsledný výstup sítě.
  
- **váhy (angl. weights)** – váhy ovlivňují, jakým způsobem se vstupní hodnoty podílí na výpočtu neuronu, každý vstup je do neuronu je vážen pomocí váhového faktoru;
- **práh (bias)** – představuje hodnotu, která umožňuje posunout aktivační funkci neuronu;
- **sumační funkce** – pro každý neuron se provádí vážená suma vstupů, často s přidáním biasu;
- **aktivační funkce** – tato funkce rozhoduje o aktivaci neuronu a definuje jeho výstup, nejčastěji používané aktivační funkce a jejich grafické znázornění:
  - **dávky (angl. batches)** – vstupní datová sada;
  - **trénovací data** – obsahují vstupy spolu s odpovídajícími očekávanými výstupy;
  - **učení (angl. learning)** – proces, během kterého síť zlepšuje svoji schopnost generovat správné výstupy, kdy adaptuje své váhy a parametry na základě trénovacích dat;
  - **epocha** – jedna fáze průchodu všemi trénovacími daty během učení sítě (Doležel, 2016).



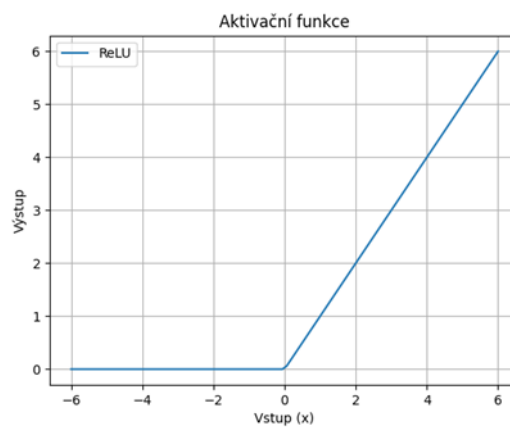
Obr. 5.3 - Aktivační funkce Softmax



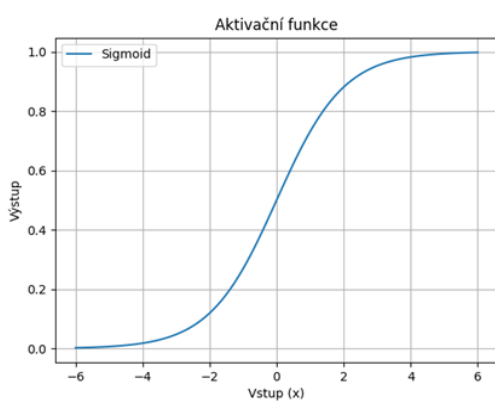
Obr. 5.4 - Aktivační funkce Tanh



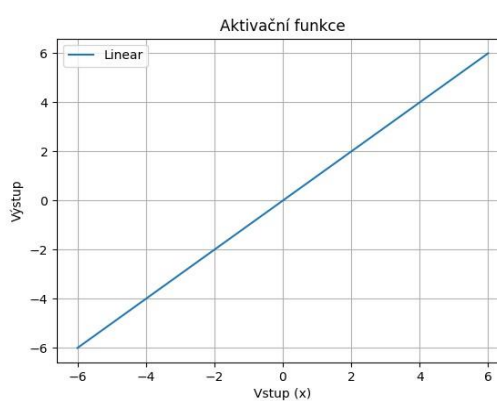
Obr. 5.5 - Aktivační funkce lineárně saturovaná



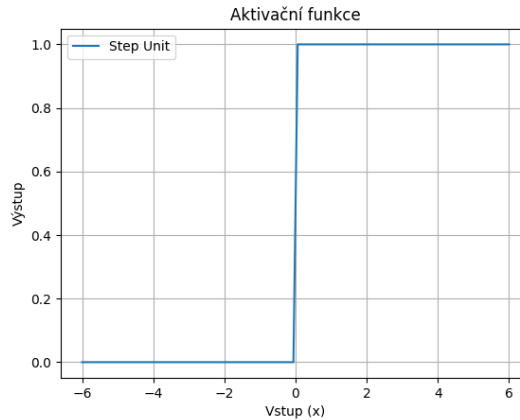
Obr. 5.6 - Aktivační funkce ReLU



Obr. 5.7 - Sigmoidní aktivační funkce



Obr. 5.8 - Lineární aktivační funkce



Obr. 5.9 - Skoková aktivační funkce

## 5.2 PROCES UČENÍ

Schopnost učení se je základní a klíčovou vlastností umělých neuronových sítí. Učení představuje dynamický proces, během něhož se upravují nastavitelné parametry sítě tak, aby dosáhla požadované funkčnosti. Tento proces řídí algoritmus učení, který určuje, jakým způsobem se parametry mění (Jiang, 2020).

### 5.2.1 Hebbův zákon učení

Hebbovo učení, popsané Donaldem Hebbem v roce 1949, představuje základní princip fungování neuronových sítí. Tato teorie uvádí, že pokud jsou dva neurony aktivovány současně, dochází k posílení synaptických spojení mezi nimi. Naopak, pokud jsou aktivovány v opačných časech, toto spojení se oslabuje. Tento proces umožňuje neuronové síti učit se z dat a vytvářet komplexní vzorce chování.

Hebbovo učení tvoří základ algoritmů zpětného šíření chyby, které se používají k trénování neuronových sítí pro širokou škálu úkolů, jako je rozpoznávání obrazu, strojový překlad, predikce atd. Smyslem Hebbova pravidla učení je upravit váhy sítě úměrně k součinu hodnot vstupů a výstupů sítě. Pravidlo můžeme popsat následujícím vzorcem:

$$\Delta w_{ij} = \eta y_i x_j \quad (5.1)$$

kde  $\Delta w_{ij}$  je změna váhy spojení mezi neuronem  $i$  a neuronem  $j$ ,  $\eta$  je koeficient učení (konstanta),  $y$  je aktivace neuronu  $i$  a  $x$  je aktivace neuronu  $j$ . Nevýhodou Hebbova pravidla je, že váhy mohou růst, pokud nejsou nějakým způsobem normalizována, což vede k nekonečnému procesu učení (Kuriscak, 2015).

## 5.2.2 Učení s učitelem

Učení s učitelem (angl. Supervised learning) představuje typ strojového učení a je spojeno s přítomností supervizora jako učitele. Tento algoritmus učení spočívá v trénování modelu z předem označených dat. Tato data tvoří tréninkovou sadu, u kterých je známa správná klasifikace nebo cílová hodnota. Na základě analýzy těchto dat se algoritmus učí rozpoznávat vzorce a vytváří tak model, který mu umožňuje dělat predikce o datech nových (Alloghani, 2020).

## 5.2.3 Učení bez učitele

Učení bez učitele (angl. Unsupervised learning) je typ strojového učení, které se učí z neoznačených dat. Data nemají žádnou předem existující klasifikaci ani nejsou označena do kategorií. Úkolem algoritmu je seskupovat nezařazená data podle podobností, vzorů a rozdílů bez předchozího trénování dat.

Na rozdíl od předem uvedeného algoritmu zde není žádný učitel. Stroj je tedy omezen na to, aby sám našel skrytou strukturu v neoznačených datech (Alloghani, 2020).

## 5.3 KONVOLUČNÍ NEURONOVÉ SÍTĚ

Konvoluční neuronové sítě (z angl. Convolutional Neural Networks, CNN) jsou analogií tradičních ANN, protože se skládají z neuronů, které se samy optimalizují učením. Každý neuron stále přijímá vstup a provádí operaci (např. skalární součin následovaný nelineární funkcí) – základ mnoha ANN. Od vstupních nezpracovaných obrazových vektorů až po konečný výstup v podobě skóre třídy bude celá síť stále vyjadřovat jedinou funkci vnímání skóre (váhu). Poslední vrstva bude obsahovat ztrátové funkce spojené se třídami (O'Shea, 2015).

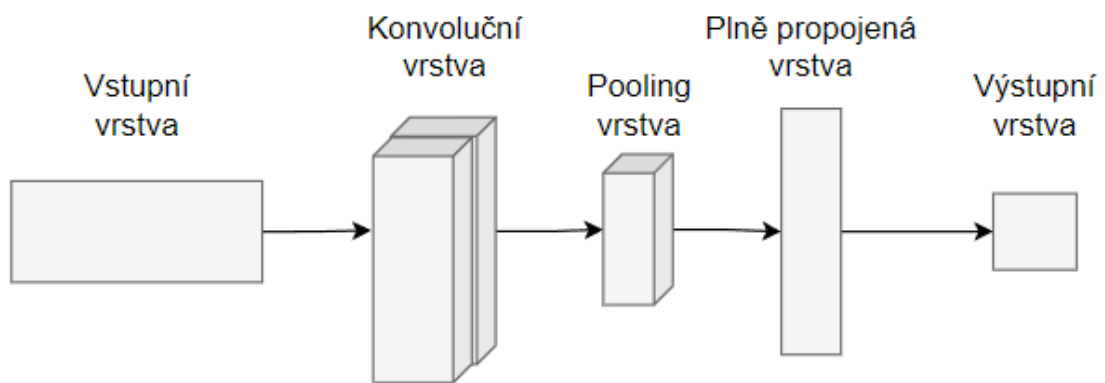
### 5.3.1 Architektura CNN

Architektura CNN se skládá z následujících vrstev:

- **Konvoluční vrstva** – Představuje samotné srdce sítě a vykonává operaci známou jako konvoluce. Konvoluce zahrnuje posouvání malého filtru (jádra) přes vstupní obrazová data a následný výpočet součinu pixelů pod filtrem násobených odpovídajícími váhami naučenými během tréninku filtru. Výsledkem tohoto procesu je aktivační mapa, která zachycuje míru výskytu vzoru odpovídajícího filtru v různých částech obrazu. Váhy

naučené během procesu učení sítě hrají klíčovou roli v schopnosti sítě extrahovat relevantní obrazové prvky;

- **Slučovací (Pooling) vrstva** – Vrstva slouží k redukci dimenzionality aktivační mapy z předchozí vrstvy. Toho je dosaženo agregací pixelů v malých oknech, přičemž každé okno je nahrazeno hodnotou reprezentující danou oblast. Pooling redukuje počet parametrů sítě a pomáhá předcházet jevu známému jako přeučení. Nadměrný počet parametrů v síti může vést k tomu, že se během učení síť „naučí“ spíše šum v datech než obecné charakteristiky, což vede k zhoršení výkonu na testovacích datech;
- **Plně propojená vrstva** – Tato vrstva se podobá vrstvám v běžných neuronových sítích a slouží k propojení neuronů z předchozí vrstvy do jednoho vektoru. Plně propojené vrstvy se obvykle používají ke klasifikaci vstupního obrazu do předem definovaných kategorií nebo k regresi číselných hodnot (O'Shea, 2015).



Obr. 5.10 - Architektura sítě CNN, převzato z (Li, 2022)

## 5.4 REKURENTNÍ NEURONOVÉ SÍTĚ

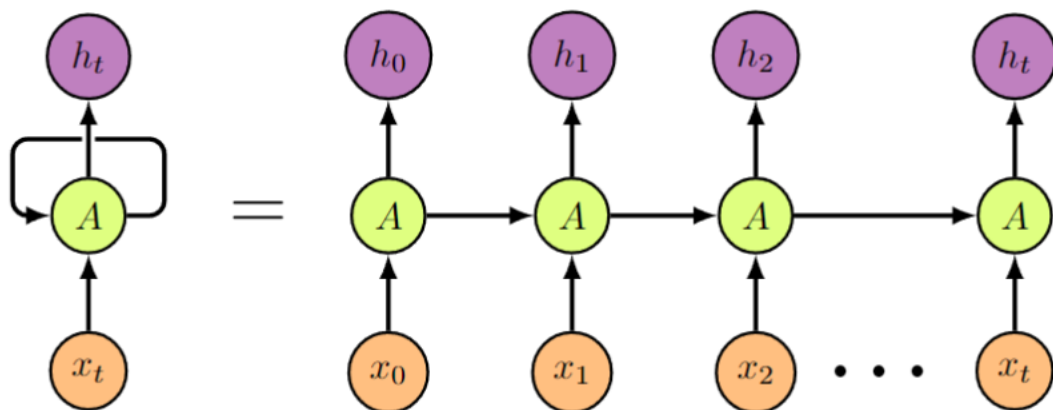
Rekurentní neuronové sítě (z angl. Recurrent neural network, RNN) jsou typem neuronových sítí, které se používají především k detekci vzorů v sekvenci dat. Takovými daty mohou být rukopis, genomy, text nebo číselné časové řady, které často vznikají v průmyslovém prostředí např. v senzorech. Jsou avšak použitelné pro obrázky, pokud je rozložíme, resp. rozdělíme na řadu políček a budeme s nimi zacházet stejně jako se sekvencí. Na vyšších úrovních nacházejí RNN uplatnění v modelování jazyka a generování textu, rozpoznávání řeči, generování popisování obrázků nebo značkování videa. RNN se od CNN známých také jako vícevrstvé perceptrony (z angl. Multilayer perceptron, MLP) liší způsobem, jakým

se informace sítě předávají. Zatímco CNN sítě předávají informace skrz síť bez použití cyklů, RNN mají cykly a předávají informace zpět do sebe, aby braly v úvahu i předchozí vstupy  $X_{0:t-1}$  a nejen aktuální vstup  $X_t$  (Schmidt, 2019).

### 5.4.1 Architektura RNN

Základní architektura RNN se skládá ze tří hlavních komponent:

- **Vstupní vrstva** – Přijímá sekvenci vstupních vektorů, jeden pro každý časový krok;
- **Skrytá vrstva** – Obsahuje rekurzivní neuronové buňky, které zpracovávají vstupní vektory a uchovávají vnitřní paměť. Každá buňka má váhy a aktivační funkci, které určují jak se vstupní vektory transformují a jak se aktualizuje vnitřní paměť;
- **Výstupní vrstva** – Generuje výstupní vektory, jeden pro každý časový okamžik. Výstupní vektory jsou obvykle závislé na vnitřní paměti skryté vrstvy a předchozích výstupních vektorech (Schmidt, 2019).



Obr. 5.11 - Architektura RNN, převzato z (Olah, 2015)

## 5.5 SÍŤ S DLOUHOU KRÁTKODOBOU PAMĚTÍ

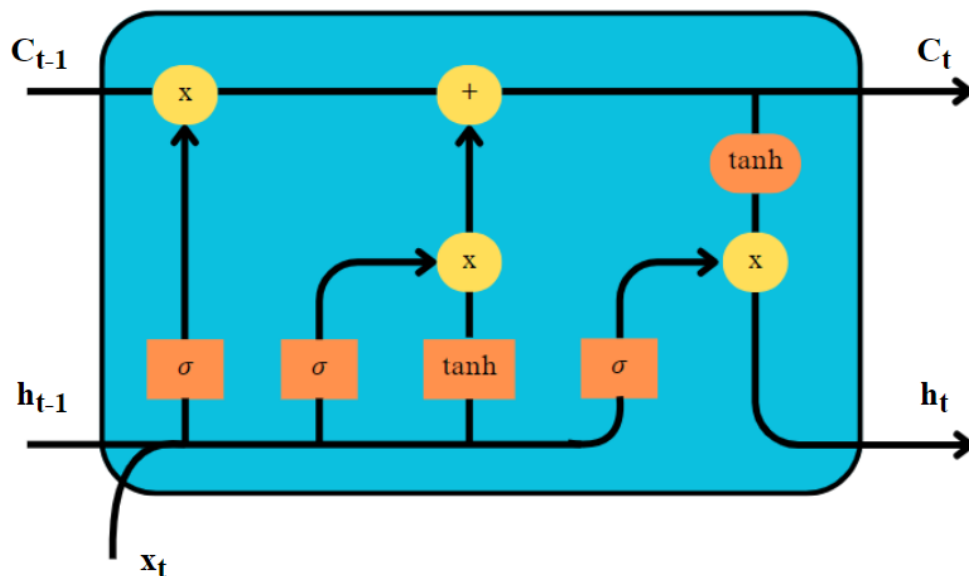
Sítě s dlouhou krátkodobou pamětí (z angl. Long Short-Term Memory, LSTM) představují specifický typ architektury RNN, jež byla vytvořena za účelem řešení stěžejní problematiky v oblasti modelování sekvencí, a to problematiky mizejícího gradientu. Tradiční RNN se potýkají s obtížemi při učení dlouhodobých závislostí v rámci sekvencí. Oproti tomu LSTM sítě překonávají tuto překážku implementací paměťových buněk, které disponují schopností selektivně uchovávat či zapomínat informace z minulých vstupů. Díky této vlastnosti se LSTM sítě stávají efektivním nástrojem pro studium a modelování sekvencí

dat, čímž se otevírá široké možnosti využití v praxi, např. predikce časových řad, rozpoznávání řeči a strojový překlad (Sak, 2014).

### 5.5.1 Architektura LSTM

Základní strukturu LSTM sítě tvoří trojice vrstev: vstupní, rekurentní LSTM a výstupní vrstva. Informace ze vstupní vrstvy vstupují do vrstvy LSTM, kde dochází k jejich zpracování v rámci rekurentních spojení. Hlavními elementárními jednotkami zodpovědnými za zpracování informací v této vrstvě jsou tzv. buňky. Každá buňka obsahuje komplexní systém bran:

- **Vstupní brána** – Řídí tok aktivací z aktuálního vstupu do paměti buňky. Určuje, které informace budou uloženy a využity pro další zpracování;
- **Zapomínající brána** – Reguluje zapomínání informací z paměti buňky. Určuje, které informace z minulých kroků sekvence jsou již relevantní a mohou být z paměti vymazány;
- **Výstupní brána** – Určuje, které informace z paměti buňky budou použity pro výpočet v daném kroku sekvence. Tyto brány plní klíčovou funkci v regulaci toku v rámci sekvence libovolné délky. Díky jejich činnosti se LSTM sítě stávají adaptivními a dokáží se selektivně zaměřit na relevantní informace v sekvenci, a to jak v krátkodobém, tak i dlouhodobém horizontu. Tato adaptibilita umožňuje efektivní učení z minulých datových bodů v sekvencích (Sak, 2014).



Obr. 5.12 - Architektura buňky LSTM, převzato z (Elmaz, 2021)

### 5.5.2 Buňka LSTM

K překonání omezení při trénování klasické RNN byla navržena buňka LSTM. Místo toho, aby výstup buňky RNN byl nelineární funkcí váženého součtu aktuálních vstupů a předchozích výstupů, používá LSTM paměťové prvky k předávání informací z minulých výstupů na aktuální výstupy. LSTM má tři řídicí signály takové, že každý z nich je nelineární funkce  $\sigma$  aktivovaná váženým součtem aktuálního vstupního pozorování  $x_t$  a předchozího skrytého stavu  $h_{t-1}$ . Zapomínající hradlo  $f_t$  rozhoduje, zda zachovat nebo zapomenout předchozí stav  $c_{t-1}$ . Vstupní hradlo  $i_t$  rozhoduje, zda se má stav LSTM aktualizovat pomocí aktuálního vstupu, či nikoli, a výstupní hradlo  $o_t$  rozhoduje, zda se skrytý stav  $h_t$  předá do další iterace, či nikoli. Nový stav  $c_t$  uložený v LSTM je součtem nového branového vstupu  $g_t$  a předchozího branového vstupu stavu  $c_{t-1}$  (Elsheikh, 2019). Matematicky můžeme brány a stavy popsat následovně:

$$i_t = \sigma(W_{ii}x_t + b_{ii} + W_{hi}h_{t-1} + b_{hi}) \quad (5.2)$$

$$f_t = \sigma(W_{if}x_t + b_{if} + W_{hf}h_{t-1} + b_{hf}) \quad (5.3)$$

$$g_t = \tanh(W_{ig}x_t + b_{ig} + W_{hg}h_{t-1} + b_{hg}) \quad (5.4)$$

$$o_t = \sigma(W_{io}x_t + b_{io} + W_{ho}h_{t-1} + b_{ho}) \quad (5.5)$$

$$c_t = f \odot c_{t-1} + i_t \odot g_t \quad (5.6)$$

$$h_t = o_t \odot \tanh(c_t) \quad (5.7)$$

kde  $x$  a  $h$  jsou vstupní (skryté) stavy,  $t$  představuje časový krok,  $\odot$  je operace násobení prvků a  $\sigma$  je sigmoidní aktivační funkce.  $g_t, f_t, i_t$  a  $o_t$  jsou buňky, zapomenuté, vstupní a výstupní brány v aktuální stavu.  $W_{xy}$  a  $b_{xy}$  představují hodnoty naučených vah a zkreslení mezi branami  $x$  a  $y$  (Elmaz, 2021).

## 5.6 UZAVŘENÁ REKURENTNÍ JEDNOTKA

Jak již bylo zmíněno, tak při zpracování dlouhých časových sekvencí v RNN dochází k jevu mizení gradientu. Proto vědci vylepšili stávající síť přidáním buněčných jednotek a tří řídicích bran ve skryté vrstvě a vytvořili tak síť LSTM. Tato síť je vhodná pro predikci dat časových řad, ale její složitá vnitřní struktura vede k dlouhé době trénování.

Uzavřená opakující se jednotka (z angl. Gated Recurrent Unit, GRU) je jednodušší variantou LSTM a byla představena v roce 2014 Cho a kol. Vnitřní struktura GRU je v porovnání s LSTM jednodušší a snadněji se trénuje, protože je do ní zapojeno méně výpočtů. Přestože jsou si GRU a LSTM velmi podobné, mají některé klíčové rozdíly (Li, 2022).

### 5.6.1 Architektura GRU

GRU používá strukturu uzavřené smyčky, která má méně parametrů než trénování LSTM a zároveň zachovává její predikční efekt. GRU obsahuje dvě brány – reset a aktualizace, zatímco LSTM disponuje třemi branami – vstupní, výstupní a zapomínající. Aktualizační brána ovládá, nakolik se informace o stavu z předchozího okamžiku zachovává v aktuálním stavu. Resetovací brána řídí, do jaké míry je aktuální stav kombinován s předchozími informacemi. Hodnoty  $z_t$  a  $r_t$  udávají, do jaké míry je informace zachována nebo ignorována. Matematicky můžeme stavy popsat následovně:

$$z_t = \sigma(W_z x_t + U_z h_{t-1}) \quad (5.8)$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1}) \quad (5.9)$$

$$\tilde{h}_t = \tanh(r_t \odot U h_{t-1} + W x_t) \quad (5.10)$$

$$h_t = (1 - z_t) \odot \tilde{h}_t + z_t \odot h_{t-1} \quad (5.11)$$

kde  $\tilde{h}_t$  je matice po spojení  $h_{t-1}$  a  $x_t$ , mapovaná na interval  $[-1, 1]$  pomocí hyperbolické tangenciální aktivační funkce ( $\tanh$ ),  $\sigma$  je sigmoidní funkce mapující hodnoty dat do intervalu  $[0,1]$  aby působil jako řídicí signál,  $W_z, W_r, W, U_z, U_r,$  a  $U$  jsou matice parametrů,  $\odot$  je skalární násobení matice (Li, 2022).



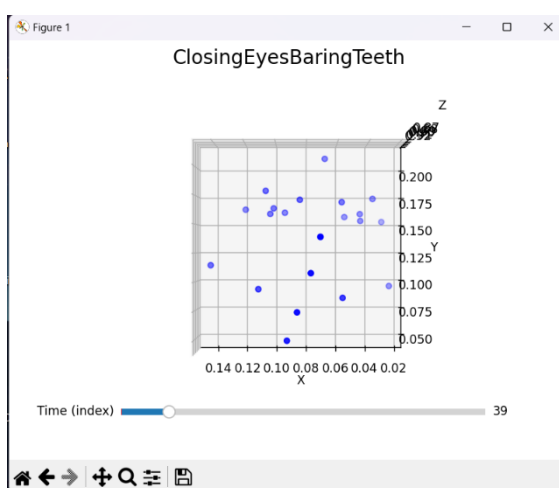
## 6 SBĚR DAT

Datový soubor využitý v této diplomové práci sleduje pacienty po operaci mozku, u kterých bývá poškozen lícní nerv a byl využit se schválením oddělení ORL Fakultní nemocnice Královské Vinohrady, kde byla měření prováděna. V době psaní této práce obsahoval datový soubor celkem 260 pacientů. Během rehabilitačního procesu pacienti plnili devět cviků zaměřených na posílení mimických svalů obličeje viz tabulka níže:

Tab. 6.1 - Přehled rehabilitačních cvičení

Rehabilitační cvičení	
1 Zvedání obočí	6 Špulení rtů
2 Mračení se	7 Vyfouknutí tváří
3 Pevné zavření očí	8 Pevné zavření očí a vycenění zubů
4 Úsměv	9 Zvednutí obočí a sevření rtů
5 Vycenění zubů	

Během provádění rehabilitačních cviků byl každý pacient snímán 3D kamerou Microsoft Kinect v2. Kamera v reálném čase analyzovala hloubkové snímky z pořízených záznamů a pomocí algoritmu založeném na modelu aktivního vzhledu z nich extrahovala obličejové rysy. Prostorové souřadnice obličejových značek, čas a název cvičení se ukládají do datových souborů pro další analýzu. Tyto značky představují 21 sledovaných identifikačních bodů obličeje, které jsou mapovány viz obr. 17. Tato data poté umožňují detailní sledování pohybů obličejových svalů v průběhu rehabilitace a hodnocení její efektivity.

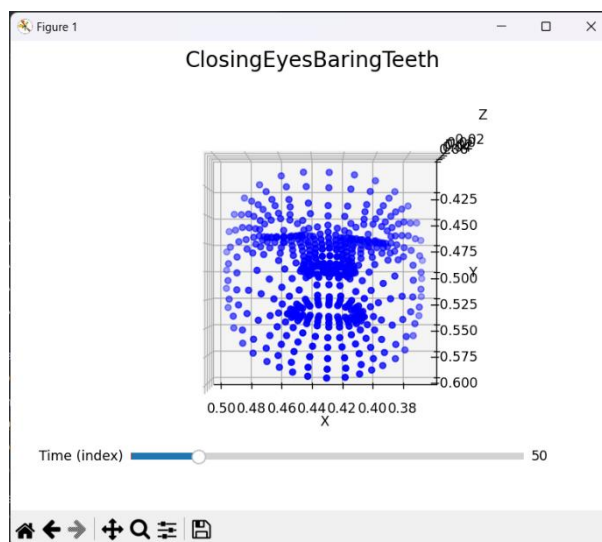


Obr. 6.1 - Vykreslení obličejových značek

## 6.1 VIZUALIZACE VE 3D

Pro 3D rekonstrukci obličeje byl využit Google MediaPipe, který je navržen pro zpracování dat v reálném čase, s nízkou latencí a vysokou propustností. Jedná se o open-source framework pro vytváření pipeline pro zpracování percepčních dat, který obsahuje algoritmy strojového učení a umožňuje budovat aplikace pro rekonstrukci obličeje nebo ruky v reálném čase. Je postaven nad knihovnou TensorFlow, která je zcela bezplatná a slouží pro vývoj aplikací strojového učení a umělé inteligence (Thierfelder, 2022)

MediaPipe obsahuje modul zvaný „FaceMesh“, který je speciálně navržen pro 3D rekonstrukci obličeje. FaceMesh se zaměřuje na 3D sledování obličeje a odhad polohy hlavy, k tomu využívá algoritmy strojového učení k detekci a sledování klíčových obličejových bodů. Tyto body zahrnují oční víčka, nos, ústa a další klíčové anatomické body (Yuan, 2020).



Obr. 6.2 - 3D rekonstrukce obličeje pomocí MediaPipe

Následující kód demonstruje načtení a vizualizaci dat o obličejích z databáze:

```
DatabaseManager.set_env(EnvironmentEnum.TEST)

measurements = DatabaseManager.get_measurements(
    filter=MeasurementFilter(
        dataset=DatasetEnum.MEDIA_PIPE_FACE,
        id_session=50),
    use_cache=True,
    verbose=True)

MeasurementPlotter.plot_3D_timeserie(measurements[5])
```

Obr. 6.3 - Kód pro načtení a vizualizaci dat z databáze

- Nastavení prostředí – Informuje *DatabaseManager* o aktuálním testovacím prostředí pro načtení správných parametrů připojení k databázi;
- Načtení dat z databáze – Zde se volá funkce *get\_measurements* pro načtení dat z databáze. Ve funkci se nachází argument *id\_session*, jehož číslo označuje a jaké sezení s pacientem se jedná;
- Vizualizace dat – K vizualizaci dat se používá třída *MeasurementPlotter*. Funkce *plot\_3D\_timeserie* vykresluje časovou řadu 3D dat obličejů. Argument *measurements[5]* označuje rehabilitační cvičení, které pacient během sezení vykonává.

## 6.2 PŘÍSTUP K DATOVÉMU SOUBORU

Získání datové sady vyžadovalo připojení k databázi, která není dostupná z vysokoškolské sítě Eduroam. Z tohoto důvodu bylo nutné využít připojení přes virtuální privátní síť (VPN). Přístup k VPN mi byl zprostředkován kolegy z Vysoké školy chemicko-technologické v Praze (VŠCHT). Po obdržení přístupových údajů jsem se k databázi připojil pomocí aplikace Cisco AnyConnect. Po aktivování VPN byly dva způsoby, jak se k použitému datasetu dostat. Prvním způsobem bylo připojení k serveru přes SQL Server Management Studio 19.



Výše uvedený blok kódu zajišťuje flexibilní konfiguraci databáze v závislosti na aktuálním prostředí. Tím umožňuje používat stejný kód pro různá prostředí (testování, vývoj, produkce) s odlišnými databázemi.

```
if env == EnvironmentEnum.DEV:
    self.db_server = "445-A336-WSRVR1.vscht.cz"
    self.db_database_name = "RehabAppDev"
elif env == EnvironmentEnum.TEST:
    self.db_server = "445-A336-WSRVR1.vscht.cz"
    self.db_database_name = "RehabAppTest"
elif env == EnvironmentEnum.DEMO:
    self.db_server = "445-A336-WSRVR1.vscht.cz"
    self.db_database_name = "RehabAppDemo"
elif env == EnvironmentEnum.PRODUCTION:
    self.db_server = "445-A336-WSRVR1.vscht.cz"
    self.db_database_name = "RehabApp"
elif env == EnvironmentEnum.LOCAL:
    self.db_server = "localhost"
    self.db_database_name = "RehabAppTest"
elif env == EnvironmentEnum.AZURE:
    self.db_server = "rehabappserverdbserver.database.windows.net"
    self.db_database_name = "RehabAppServerDB"
else:
    raise NotImplementedError(f"{env} is not configured")
```

Obr. 6.6 - Pokračování kódu pro připojení k databázi

## 7 KLASIFIKACE POMOCÍ LSTM

V této části diplomové práce jsem se zaměřil na klasifikaci pomocí LSTM sítě v programovacím jazyce Python a s využitím knihovny Keras. Jak již bylo řečeno v kapitole o LSTM, tak jsou tyto sítě efektivním nástrojem pro analýzu sekvenčních dat, v tomto případě jde tedy o klasifikaci na základě časových řad dat pohybu obličeje pacientů.

### 7.1 PŘEDZPRACOVÁNÍ DAT

Nejprve bylo nutné upravit formát vstupních dat z databáze, tak aby odpovídal požadavkům LSTM sítě. Data souřadnic pohybu obličeje pacientů s parézou lícního nervu byla rozdělena na trénovací a testovací sadu v poměru 80:20. Toto rozdělení je běžné v oblasti strojového učení a umožňuje objektivní vyhodnocení výkonu modelu na předem neviděných datech.

Před rozdělením dat byla zajištěna maximální délka časové řady v souboru. Tato informace byla použita k nastavení parametru *time\_dimension\_lenght* při vytváření objektu *MeasurementSet*. Tento parametr určuje maximální počet časových kroků, které budou brány v úvahu při zpracování dat LSTM sítí.

Následně byly z objektu *MeasurementSet* získány souřadnice pohybu (coordinates) a klasifikace parézy lícního nervu (hb\_classifications). Data byla dále rozdělena na testovací a trénovací sadu pomocí funkce *train\_test\_split* za pomoci knihovny scikit-learn. Parametr *random\_state* byl nastaven na 42 aby se zajistila reprodukovatelnost rozdělení dat při opakovaném spuštění kódu.

```
# Načtení dat z databáze
DatabaseManager.set_env(EnvironmentEnum.TEST)
measurements = DatabaseManager.get_measurements(
    filter=MeasurementFilter(
        dataset=DatasetEnum.KINECT_V2,
        healthy=False),
    use_cache=True,
    verbose=True)

max_time_length = np.max([len(measurement.time) for measurement in measurements])
measurement_set = MeasurementSet(measurements, time_dimension_length=1000)

coordinates = measurement_set.get_coordinates()
hb_classifications = measurement_set.get_hb_classifications()

# Rozdělení dat
x_train, x_test, y_train, y_test = train_test_split(*arrays: coordinates, hb_classifications,
                                                    test_size=0.2, random_state=42)
```

Obr. 7.1 – LSTM: Předzpracování dat

Data souřadnic (`x_train` a `x_test`) byla dále upravena pomocí funkce `reshape` pro zjednodušení vstupu do LSTM sítě. Nový tvar spojuje časové kroky a dimenze souřadnic do jedné dimenze, čímž zjednodušuje zpracování dat sítí. Nakonec byla klasifikace parézy lícního nervu (`y_train` a `y_test`) konvertována na one-hot vektory pomocí funkce `to_categorical` z knihovny `scikit-learn`. Funkce převádí celočíselné klasifikace na one-hot vektory s délkou odpovídající počtu tříd (`num_classes = 7`). Tyto Vektory usnadňují zpracování kategoriálních proměnných proměnných sítí. Tvar vektorů byl poté upraven pomocí funkce `reshape` pro konzistentnost s daty souřadnic.

```
# Načtení a zpracování dat
x_train = x_train.reshape(x_train.shape[0], -1, x_train.shape[2]*x_train.shape[3])
x_test = x_test.reshape(x_test.shape[0], -1, x_test.shape[2]*x_test.shape[3])

num_classes = 7
y_train_onehot = to_categorical(y_train, num_classes)
y_test_onehot = to_categorical(y_test, num_classes)

y_train_onehot = y_train_onehot.reshape(-1, num_classes)
y_test_onehot = y_test_onehot.reshape(-1, num_classes)
```

Obr. 7.2 - LSTM: Načtení a zpracování dat

## 7.2 DEFINICE A KOMPILACE MODELU

Model je definován pomocí knihovny `Keras` a skládá se ze tří vrstev:

- LSTM vrstva – Tato vrstva je jádrem modelu a je zodpovědná za zpracování sekvenčních dat. Má 128 neuronů a vrací sekvence skrytých stavů. Parametr `input_shape` určuje tvar vstupních dat (časové kroky, dimenze souřadnic). Parametr `return_sequences=True` umožňuje vrstvě vracet sekvence skrytých stavů pro další případné LSTM vrstvy;
- Vrstva `Flatten` – Vrstva slouží k transformaci z předchozí vrstvy do plochého vektoru, který je kompatibilní s následující vrstvou;
- Vrstva `Dense` – Tato vrstva je výstupní vrstvou a má počet neuronů odpovídající již uvedenému počtu tříd a používá aktivační funkci `softmax`, která generuje pravděpodobnostní rozdělení mezi 0 a 1 pro každou třídu.

Model je kompilován optimalizátorem `Adam` s nízkou rychlostí učení (`learning_rate = 0.001`) a ztrátovou funkcí `categorical_crossentropy`. Optimalizátor `Adam` se používá

pro aktualizaci vah modelu během trénování. A již zmíněná ztrátová funkce je vhodná pro klasifikační úlohy s více třídami. Kromě optimalizátoru a funkce ztráty se definují i metriky hodnocení přesností (*accuracy*), která sleduje procento správně klasifikovaných vzorků.

```
# Definice modelu
model = Sequential()
model.add(LSTM(units=128, return_sequences=True, input_shape=x_train.shape[1:]))

model.add(Flatten())
model.add(Dense(num_classes, activation='softmax'))

# Kompilace s redukovanou rychlostí učení
model.compile(optimizer=Adam(learning_rate=0.001), loss='categorical_crossentropy', metrics=['accuracy'])
```

Obr. 7.3 - LSTM: Definice modelu

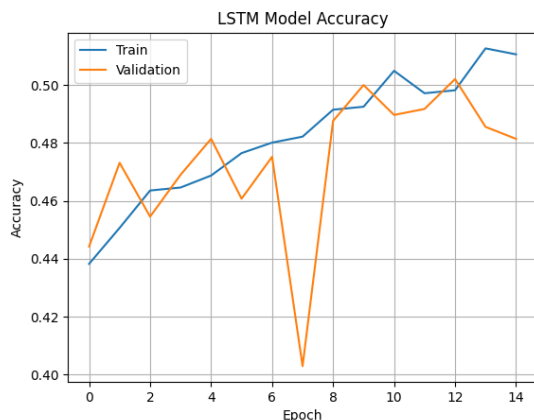
## 7.3 TRÉNOVÁNÍ MODELU

Pro první trénování modelu na datech *x\_train* a *y\_train\_onehot* jsem zvolil dobu trénování 15 epoch s velikostí dávky (*batch\_size*) 32. Funkce *fit* vrací historii trénování, která obsahuje o ztrátě přesnosti a přesnosti v trénovacích a validačních datech během průběhu jednotlivých epoch. Pro uložení vah s nejlepší přesností validace bylo využito zpětné volání (callback) *ModelCheckpoint*. To umožňuje pozdější načtení nejlépe trénovaného modelu pro testování nebo nasazení.

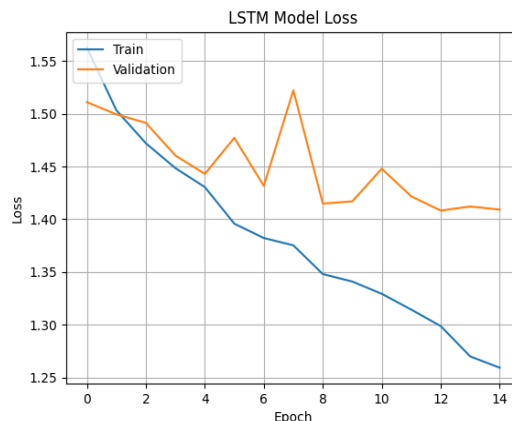
```
# Zpětné volání kontrolního bodu
checkpoint = ModelCheckpoint(filepath='best_weights.h5', monitor='val_accuracy',
                             save_best_only=True)

# Trénování modelu
history = model.fit(x_train, y_train_onehot,
                   epochs=15,
                   batch_size=32,
                   callbacks=[checkpoint],
                   validation_data=(x_test, y_test_onehot))
```

Obr. 7.4 - LSTM: Trénování modelu



Obr. 7.5 - LSTM: Validace pro experiment č. 1



Obr. 7.6 - LSTM: Ztráta pro experiment č. 1

První trénování s již zmíněným nastavením proběhlo s úspěšností validace 46,8% a ztrátou 140,9%. Průběhy pro úspěšnost validace a ztrátu:

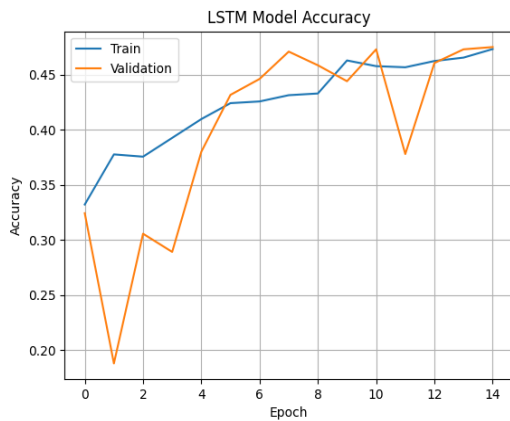
Pro druhý experiment jsem do kódu přidal dvě regularizační techniky: - *BatchNormalization* a *Dropout*. *BatchNormalization* normalizuje vstup do každé jednotlivé vrstvy v síti na jednotný rozsah a zrychluje trénování sítě a zlepšuje její stabilitu. *Dropout* náhodně deaktivuje určité neurony v každé vrstvě během tréninku s danou pravděpodobností, v tomto případě 0,5. Dále byla přidána Dense vrstva s 128 neurony, to slouží k dalšímu zpracování a redukci dimenzionality dat před finální klasifikací. Počet epoch zůstal nezměněn.

```
# Definice modelu
model = Sequential()
model.add(LSTM(units=128, return_sequences=True, input_shape=x_train.shape[1:]))
model.add(BatchNormalization())
model.add(Dropout(0.5))
model.add(Dense(128))
```

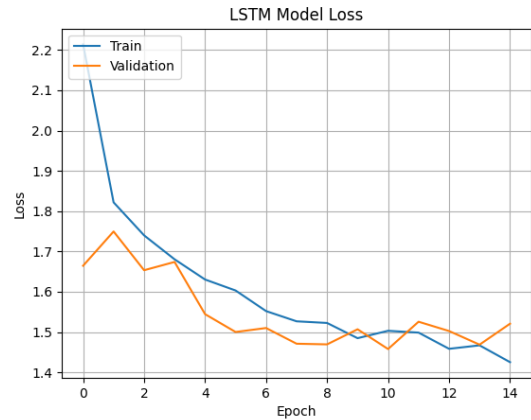
Obr. 7.7 - LSTM: Definice modelu pro experiment č. 2

Úspěšnost validace se zlepšila na 47,5 %, nicméně ztráta byla v tomto průběhu horší 152,1 %.

Pro třetí experiment byla přidána druhá LSTM a Dense vrstva, dále *BatchNormalization*, *Dropout* (0.2). Počet neuronů v nových vrstvách jsem ponechal na 128. Počet epoch byl zvýšen na 30. Trénování proběhlo s validací 46,9 % a se ztrátou 145,6 %.

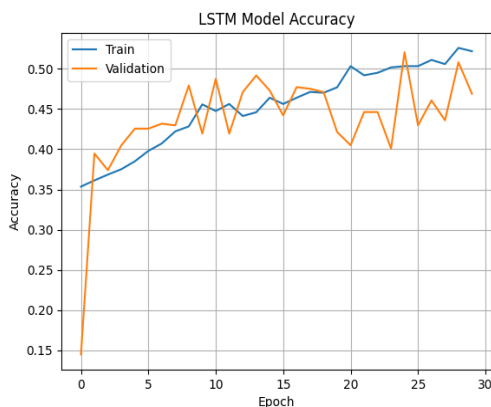


Obr. 7.8 - LSTM: Validace pro experiment č. 2

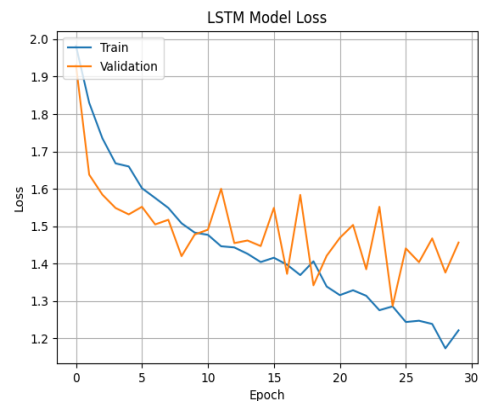


Obr. 7.9 - LSTM: Ztráta pro experiment č. 2

Pro tento čtvrtý experiment jsem kód rozšířil o dvě techniky snižování rychlosti učení pro optimalizaci sítě. První technikou je *LearningRateScheduler*, ta umožňuje dynamicky upravovat rychlost učení během tréninku. V tomto případě se rychlost učení snižuje o 10 % po každé epoše ( $0,001 * 0,9^{**}epoch$ ). Cílem je zpomalit trénování v pozdějších fázích, aby se zabránilo overfittingu a dosáhlo se optimálního zobecnění. *ReduceLROnPlateau* je druhou technikou a monitoruje validační ztrátu a snižuje rychlost učení, pokud se ztráta po dobu zadaného počtu epoch nezlepší. V tomto případě se rychlost učení snižuje o 30 % po 5 epochách bez zlepšení validační ztráty. Tato technika pomáhá zabránit uvíznutí v lokálních minimech a usnadňuje nalezení optimálního řešení. Obě techniky se přidávají do sekce zpětného volání v trénování modelu.



Obr. 7.10 - LSTM: Validace pro experiment č. 3



Obr. 7.11 - LSTM: Ztráta pro experiment č. 3

```

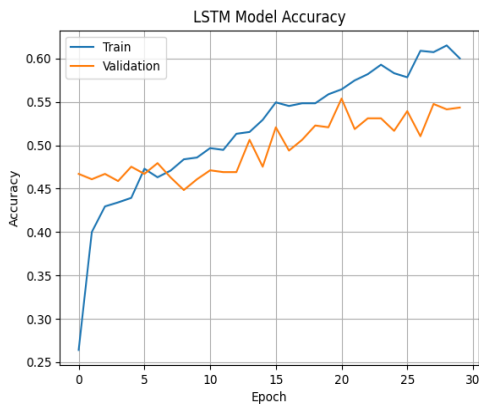
# Definice snížení rychlosti učení
lr_scheduler = LearningRateScheduler(lambda epoch: 0.001 * 0.9**epoch)
rlrop = ReduceLROnPlateau(monitor='val_loss', factor=0.3, patience=5)

# Trénování modelu
history = model.fit(x_train, y_train_onehot,
                    epochs=30,
                    batch_size=32,
                    callbacks=[checkpoint, lr_scheduler, rlrop],
                    validation_data=(x_test, y_test_onehot))

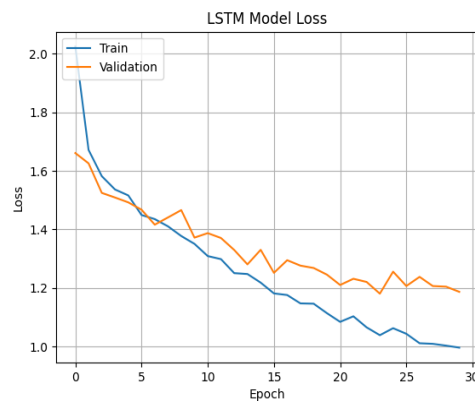
```

Obr. 7.12 - LSTM: Trénování modelu pro experiment č. 4

Úspěšnost validace se po trénování modelu zvýšila na 54,3% a ztrátou 122,1%. Provedené rozšíření kódu pro snížení rychlosti učení mělo pozitivní dopad na trénování modelu.

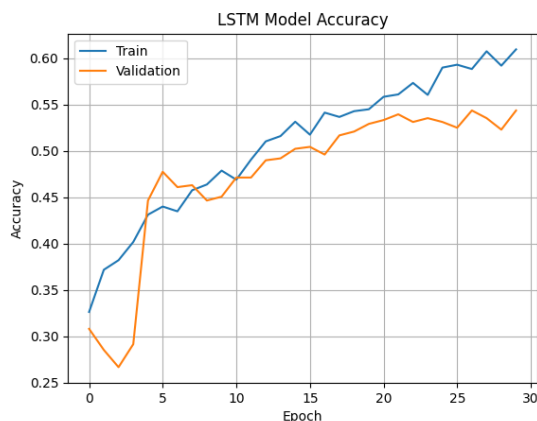


Obr. 7.13 - LSTM: Validace modelu pro experiment č. 4

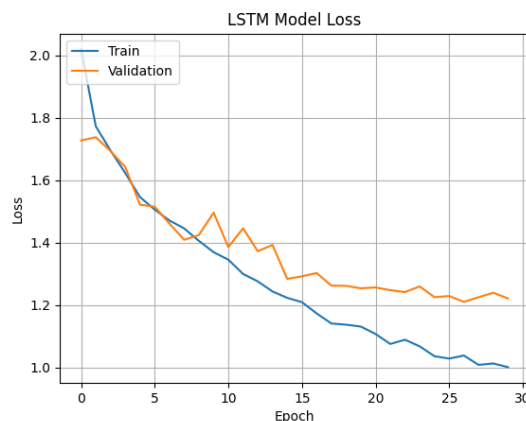


Obr. 7.14 - LSTM: Ztráta modelu pro experiment č. 4

V pátém experimentu jsem přidal již třetí LSTM a Dense vrstvu, opět po 128 neuronech. S vrstvami se přidali i techniky pro regularizaci *BatchNormalization* a *Dropout* (0.1).



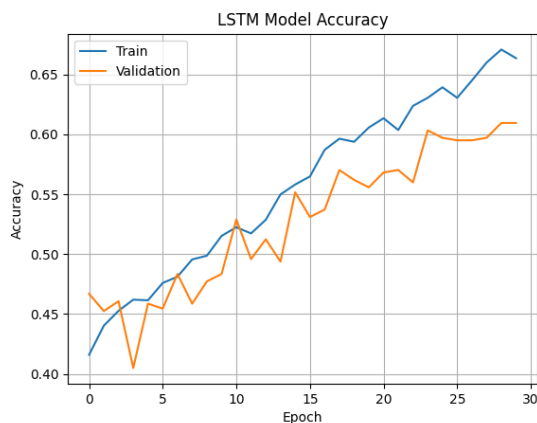
Obr. 7.15 - LSTM: Validace modelu pro experiment č. 5



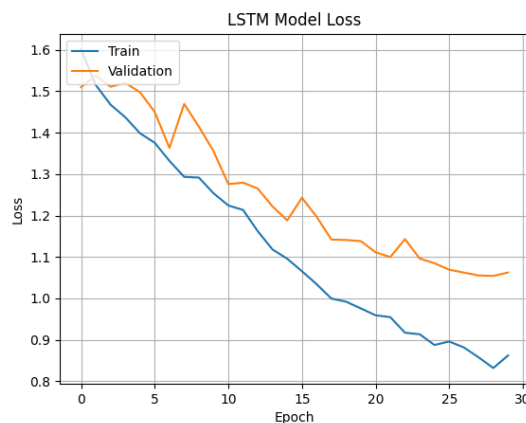
Obr. 7.16 - LSTM: Ztráta modelu pro experiment č. 5

Průběh validace je zůstal stejný jako u předchozího experimentu, tedy 54,3 %. Nicméně ztráta se snížila na 118,7 %.

Pro šestý experiment jsem ponechal počet trénovacích epoch na 30 a upravil rychlost učení, se kterou operuje optimalizátor *Adam* (0,0001).

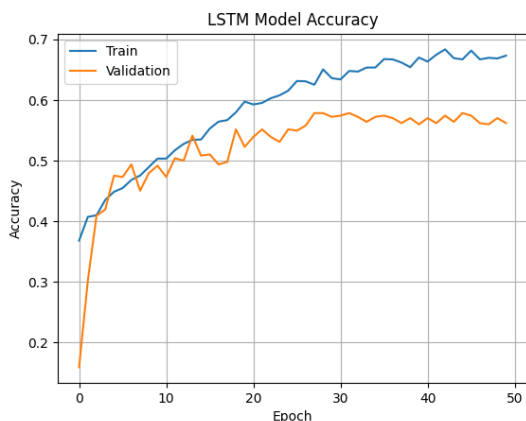


Obr. 7.17 - LSTM: Validace modelu pro experiment č. 6

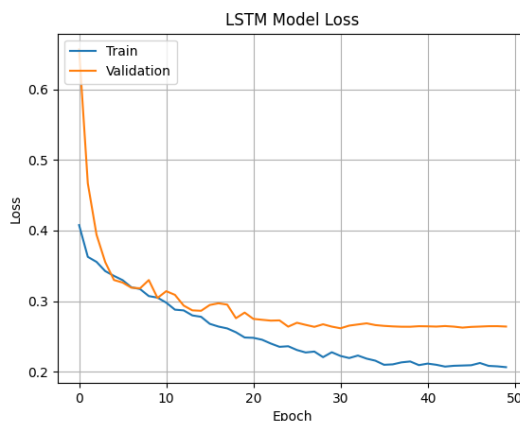


Obr. 7.18 - LSTM: Ztráta modelu pro experiment č. 6

Průběh validační množiny po trénování dosáhl 61 % se ztrátou 106,3 %.



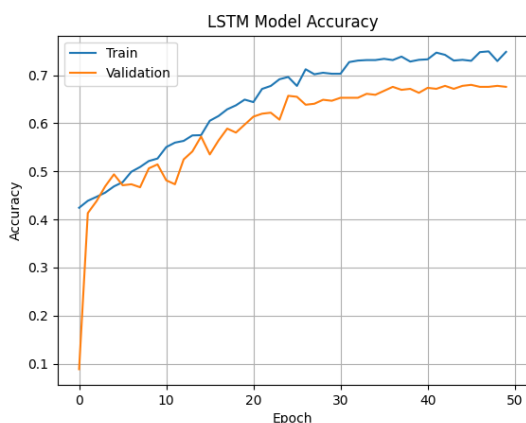
Obr. 7.19 - LSTM: Validace modelu pro experiment č. 7



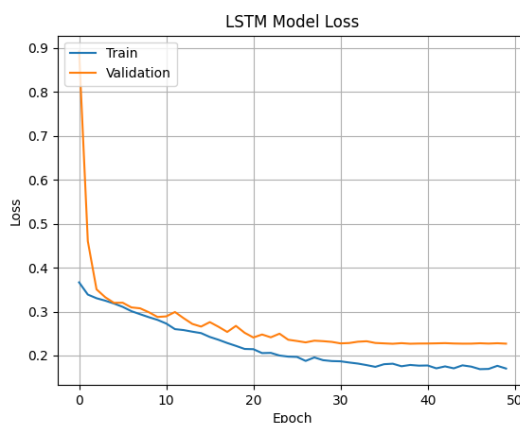
Obr. 7.20 - LSTM: Ztráta modelu pro experiment č. 7

Pro trénovací průběh sedmého experimentu byl zvýšen počet epoch na 50 a byla změněna ztrátová funkce na *binary\_crossentropy*. Tato funkce vypočítává míru chyby mezi predikcí modelu a skutečnou binární hodnotou a je vhodná pro binární klasifikaci, kde existují pouze dva možné výstupy.

Úspěšnost trénování validační množiny byla oproti minulému experimentu nižší 56,2 %, naopak ztráta se dostala na 26,4 %. Ztrátová funkce využitá pro tento experiment bude použita i nadále.



Obr. 7.21 - LSTM: Validace modelu pro experiment č. 8



Obr. 7.22 - LSTM: Ztráta modelu pro experiment č. 8

Pro osmý průběh trénování byla odstraněna regularizace *BatchNormalization* ze třetí vrstvy. S cílem porovnat výsledky s přechozím průběhem.

Výsledkem tohoto experimentu je zlepšení validace na 67,7 % a ztráty na 22,7 %. Odstranění regularizační techniky mělo pozitivní dopad na trénování sítě.

## 8 KLASIFIKACE POMOCÍ GRU

V této kapitole diplomové práce se zaměřuji na klasifikaci dat pohybu obličeje pacientů s využitím GRU sítě. Vzhledem k tomu, že GRU sítě sdílí s LSTM sítěmi podobnou architekturu a data pro klasifikaci byla předzpracována identickým způsobem jako v kapitole [7.1](#), bude kapitola o samostatném předzpracování dat vynechána. Je pravděpodobné, že se v průběhu trénování dosáhne srovnatelných výsledků.

### 8.1 DEFINICE A KOMPILACE MODELU

Sít' používá *Sequential* model z knihovny Keras pro postupné přidávání vrstev. Model je tvořen třemi vrstvami s následujícími vlastnostmi:

- GRU vrstva – Obsahuje 128 neuronů a vrací sekvence skrytých stavů (výstup pro každá prvek v sekvenci);
- Flatten vrstva – Vrstva slouží k převodu sekvencí skrytých stavů z GRU vrstvy na jednorozměrný vektor;
- Dense vrstva – Jedná se o poslední vrstvu před trénováním modelu a slouží pro finální klasifikaci. Disponuje 7 neurony, které odpovídají počtu tříd, do kterých chceme data zařadit. *Softmax* aktivační funkce převádí výstupy na pravděpodobnostní rozdělení nad těmito třídami.

Model je opět kompilován optimalizátorem Adam se stejnou počáteční rychlostí učení, tedy 0,001. Ztrátová funkce byla již od začátku trénování změněna na *binary\_crossentropy*.

```
# Definice modelu
model = Sequential()
model.add(GRU(units=128, return_sequences=True, input_shape=x_train.shape[1:]))

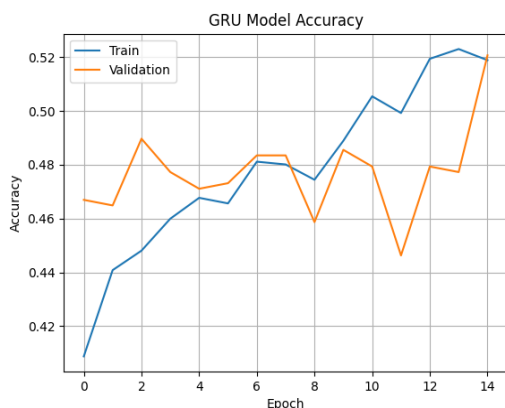
model.add(Flatten())
model.add(Dense(num_classes, activation='softmax'))

# Kompilace s redukovanou rychlostí učení
model.compile(optimizer=Adam(learning_rate=0.001), loss='binary_crossentropy', metrics=['accuracy'])
```

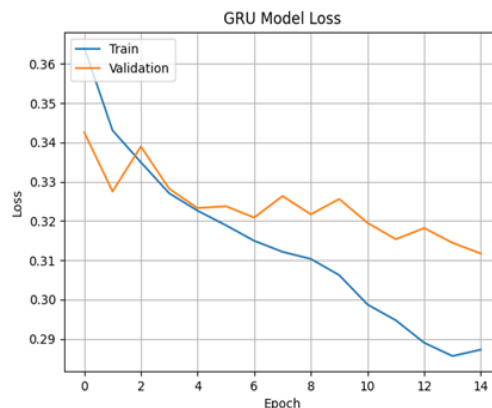
Obr. 8.1 - GRU: Definice modelu

### 8.2 TRÉNOVÁNÍ MODELU

Pro první trénování sítě jsem postupoval stejně jako v případě trénování předchozí sítě. Trénování probíhalo s třemi základními vrstvami po dobu 15 epoch a s velikostí dávky 32.



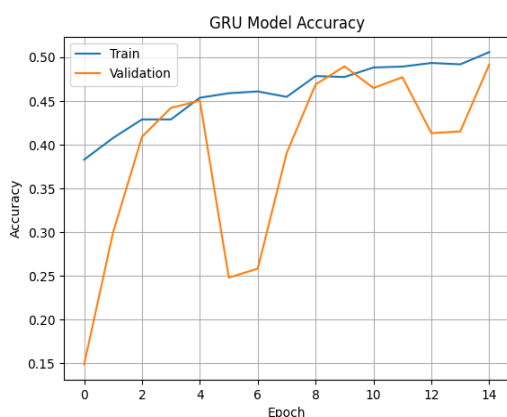
Obr. 8.2 - GRU: Validace modelu pro experiment č. 1



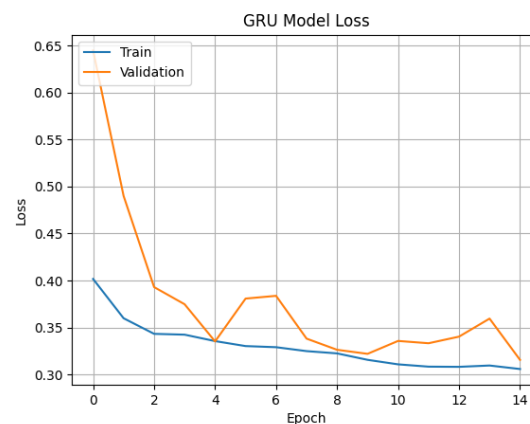
Obr. 8.3 - GRU: Ztráta modelu pro experiment č. 1

Průběh validace při prvním trénování je 52 % se ztrátou 31,1 %. V porovnání s úspěšností validace při prvním testování LSTM je tento průběh lepší o 5,2 %.

Do druhého průběhu byly implementovány regularizační techniky (*BatchNormalization*, *Dropout*) a zároveň byla přidána Dense vrstva se 128 neurony. Validace se po ukončení trénování zastavila na 49,1 % a ztráta na 31,5 %.

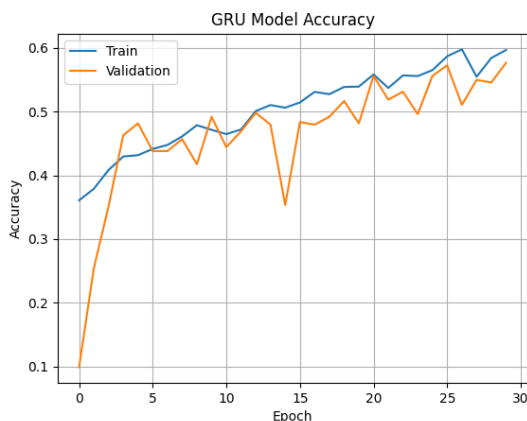


Obr. 8.4 - GRU: Validace modelu pro experiment č. 2

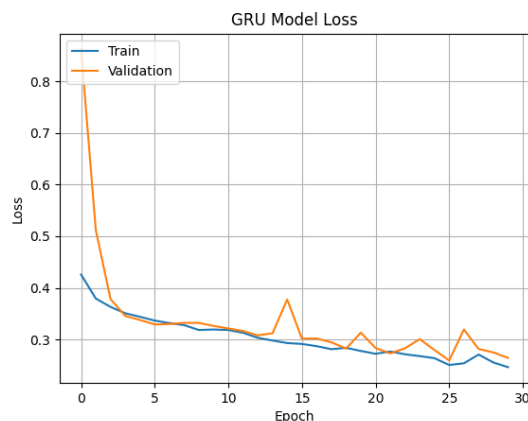


Obr. 8.5 - GRU: Ztráta modelu pro experiment č. 2

V rámci třetího experimentu jsem rozšířil architekturu sítě o druhou GRU a Dense vrstvu, obě s 128 neurony, a zároveň jsem prodloužil dobu trénování na 30 epoch. Pro optimalizaci výsledků jsem dále implementoval regularizační techniky (*Dropout = 0.3*).

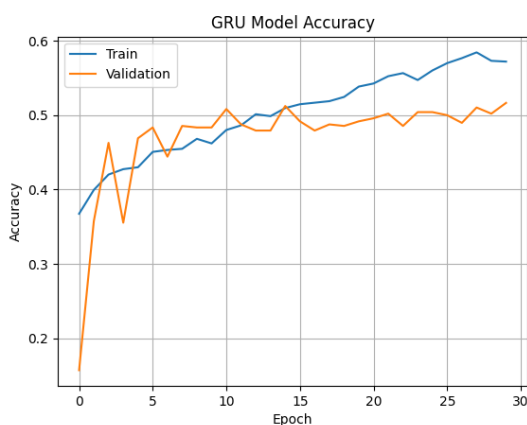


Obr. 8.6 - GRU: Validace modelu pro experiment č. 3

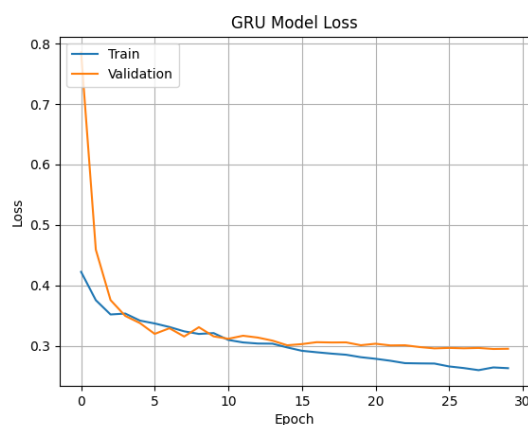


Obr. 8.7 - GRU: Ztráta modelu pro experiment č. 3

Během trénování modelu došlo k zastavení validace na 57,6 % a 26,4 % ztrátě.

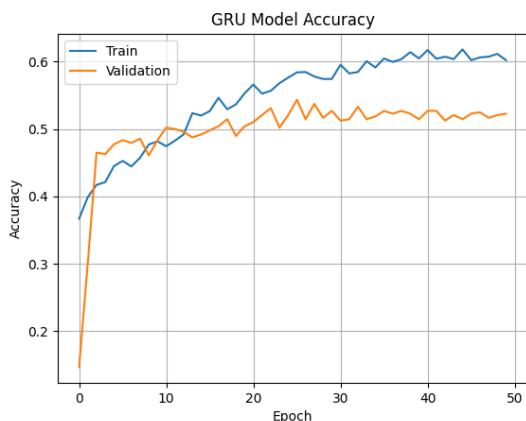


Obr. 8.8 - GRU: Validace modelu pro experiment č. 4

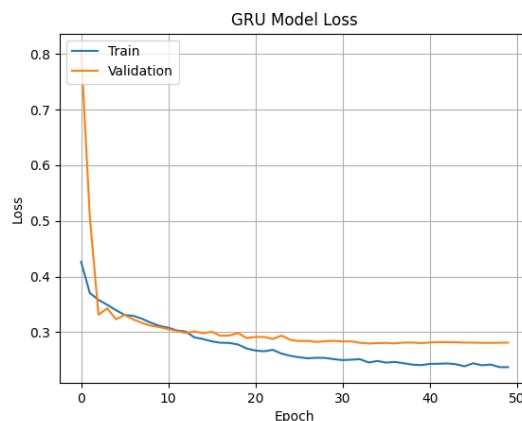


Obr. 8.9 - GRU: Ztráta modelu pro experiment č. 4

Pro čtvrtý experiment byly přidány techniky zpomalení rychlosti učení *LearningRateScheduler* a *ReduceLROnPlateau*. Výsledek trénování validace po 30 epochách byl 51,6 % se ztrátou 29,5 %.

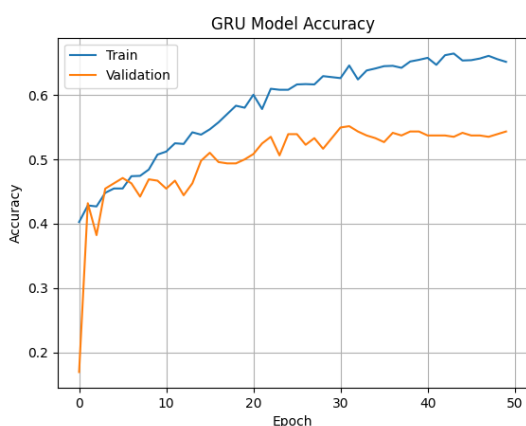


Obr. 8.10 - GRU: Validace modelu pro experiment č. 5

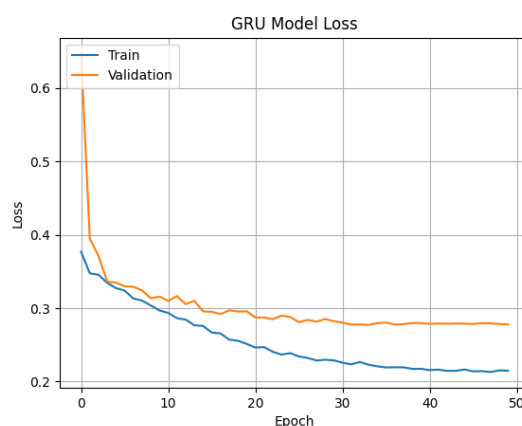


Obr. 8.11 - GRU: Ztráta modelu pro experiment č. 5

V pátém experimentu jsem zvýšil počet epoch na 50 a přidal třetí GRU a Dense vrstvu, opět po 128 neuronech. Dále byly přidány regularizační techniky ( $Dropout=0.2$ ). Trénování validační množiny dosáhlo 52,6 % se ztrátou 28,2 %.



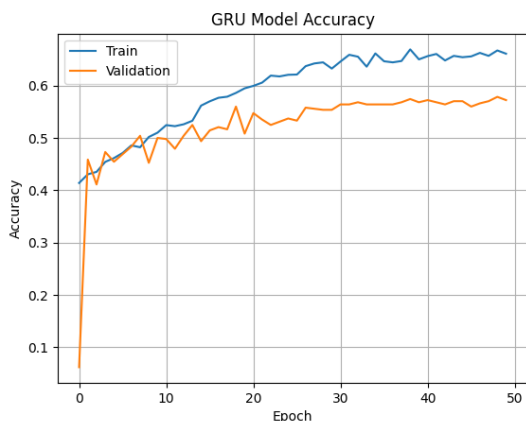
Obr. 8.12 - GRU: Validace modelu pro experiment č. 6



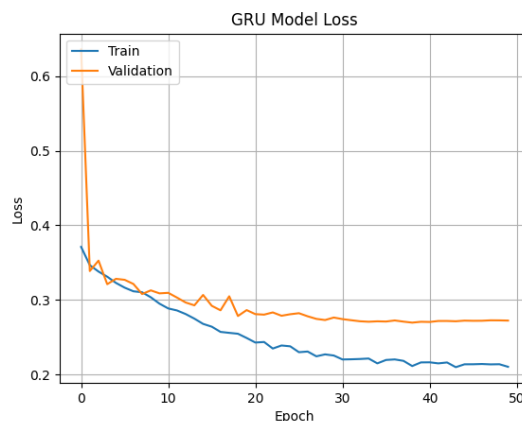
Obr. 8.13 - GRU: Ztráta modelu pro experiment č. 6

Minulý experiment nepřinesl uspokojivé výsledky, proto jsem v tomto experimentu odstranil regularizační techniku *BatchNormalization* z nově přidané třetí vrstvy. Výsledkem tohoto experimentu je zlepšení validace na 54,3 % a ztráty na 27,7 %.

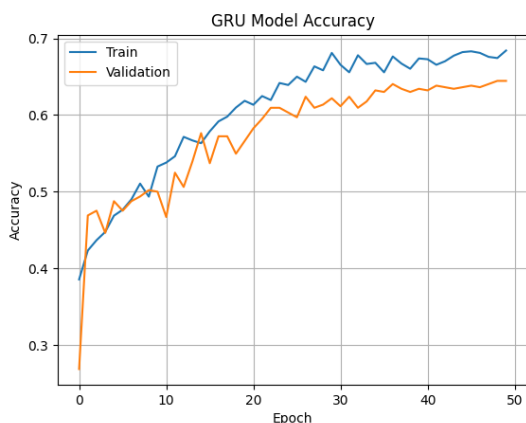
V sedmém experimentu jsem odstranil druhou regularizační techniku *Dropout* ze třetí vrstvy s cílem pokusit se o zlepšení validace jako v experimentu 6. Výsledkem toho experimentu je zlepšení validace na 57,2 % a ztráty 27,2 %. Odstranění techniky *Dropout* mělo pozitivní dopad na trénování dat.



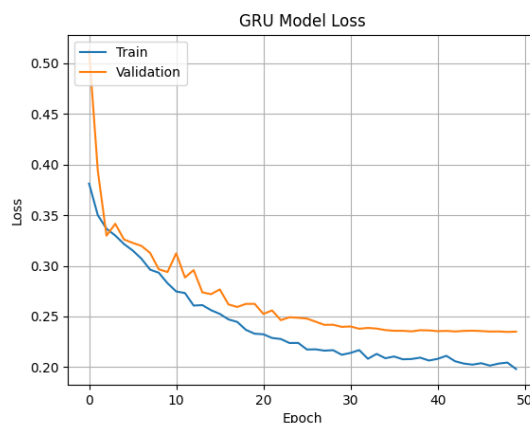
Obr. 8.14 - GRU: Validace modelu pro experiment č. 7



Obr. 8.15 - GRU: Ztráta modelu pro experiment č. 7



Obr. 8.16 - GRU: Validace modelu pro experiment č. 8



Obr. 8.17 - GRU: Ztráta modelu pro experiment č. 8

Pro osmý experiment jsem použil jsem rozhodl použít jiný optimalizátor, a to *RMSprop*. Jedná se o adaptivní optimalizátor, který automaticky upravuje rychlost učení pro každý parametr během trénování. To může vést k plynulejšímu a stabilnějšímu trénování oproti klasickým optimalizátorům s pevnou rychlostí učení. Rychlost učení byla ponechána na hodnotě 0,0001. Výsledkem je pozitivní nárůst validace na 64,4 % a snížení ztráty na 23,4 %. Implementace *RMSprop* mělo pozitivní dopad na trénování sítě.

## 9 KLASIFIKACE POMOCÍ CNN

Tato kapitola diplomové práce je zaměřena na použití CNN sítě pro klasifikaci dat obličeje pacientů, s využitím knihovny Keras.

### 9.1 PŘEDZPRACOVÁNÍ DAT

Vstupní data ( $x_{train}$ ,  $x_{test}$ ) reprezentují souřadnice pohybu obličeje pacientů. Data jsou strukturována jako trojrozměrné pole ve tvaru: počet bodů, počet časových kroků, počet souřadnic (3 souřadnice na bod). Pro CNN síť je nutné data upravit do tvaru, který vyžaduje knihovna Keras. K tomu slouží funkce `np.transpose`, která upravuje pořadí dimenzí. Transpozice prohodí pořadí časových kroků a bodů, a přesune souřadnice na poslední pozici, čímž dosáhneme požadovaného tvaru. Výsledný tvar po transpozici (1000, 21, 3), což znamená 1000 časových kroků, 21 bodů a 3 souřadnice (X, Y, Z) pro každý bod v časovém kroku.

Cílové proměnné ( $y_{train}$  a  $y_{test}$ ) reprezentující diagnózy parézy lícního nervu se převedou na one-hot vektory pomocí funkce `to_categorical` z knihovny scikit-learn.

```
# Předzpracování dat
input_shape = (1000, 21, 3)
x_train = np.transpose(x_train, axes=(0, 2, 1, 3)) # Reshape to (samples, time_steps, spatial_dim, channels)
x_test = np.transpose(x_test, axes=(0, 2, 1, 3))

y_train_onehot = to_categorical(y_train, num_classes=7)
y_test_onehot = to_categorical(y_test, num_classes=7)
```

Obr. 9.1 - CNN: Předzpracování dat

### 9.2 DEFINICE A KOMPILACE MODELU

Model CNN je implementován s využitím knihovny Keras a sestává se z následujících klíčových vrstev:

- Konvoluční vrstva (Conv2D) – Tato vrstva je jádrem CNN modelu a používá se k extrahování prostorových vlastností z vstupních dat. V kódu je použita vrstva `Conv2D` s 32 filtry o velikosti jádra 3x3 a aktivační funkcí ReLU. Filtry jsou naučené váhové vektory, které se posouvají po vstupních datech a extrahují nízkoúrovňové rysy. Velikost jádra určuje oblast vstupních dat, kterou bude filtr zpracovávat najednou. Aktivační funkce ReLU zavádí nelinearitu do modelu a umožňuje mu naučit se složitější vztahy v datech;

- Flatten vrstva – Vrstva transformuje výstup z konvoluční vrstvy do jednorozměrného pole. Je nezbytná před plně propojenou vrstvou, která vyžaduje vektorový vstup;
- Výstupní vrstva (Dense) – Plně propojená vrstva s 7 neurony slouží jako klasifikační vrstva. Počet neuronů odpovídá počtu tříd, do kterých chceme data zařadit.

Nakonec je model zkompileován pomocí optimalizačního algoritmu Adam (*learning\_rate=0,001*) a funkce *categorical\_crossentropy*.

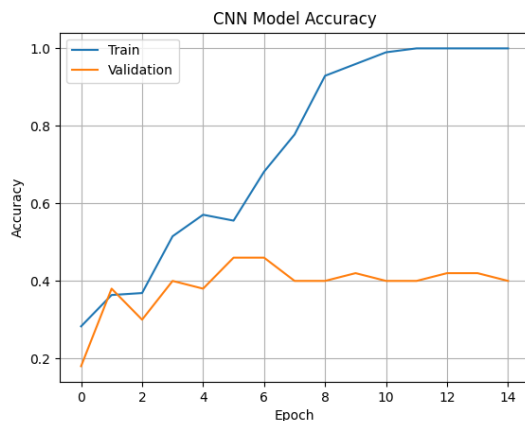
```
# Definice modelu
model = Sequential()
model.add(Conv2D(filters=32, kernel_size=(3, 3), strides=2, activation='relu'))

model.add(Flatten())
model.add(Dense(units=6, activation='softmax'))

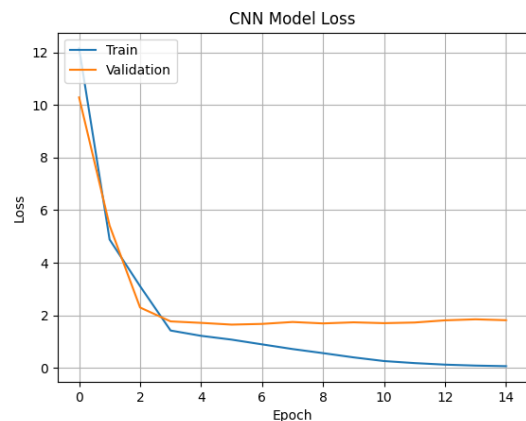
model.compile(optimizer=Adam(learning_rate=0.001), loss='categorical_crossentropy', metrics=['accuracy'])
```

Obr. 9.2 - CNN: Definice modelu

### 9.3 TRÉNOVÁNÍ MODELU



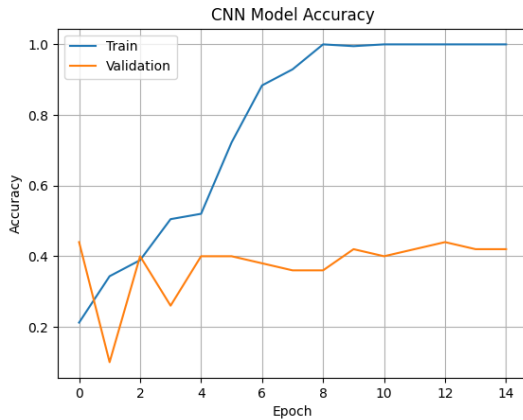
Obr. 9.3 - CNN: Validace modelu pro experiment č. 1



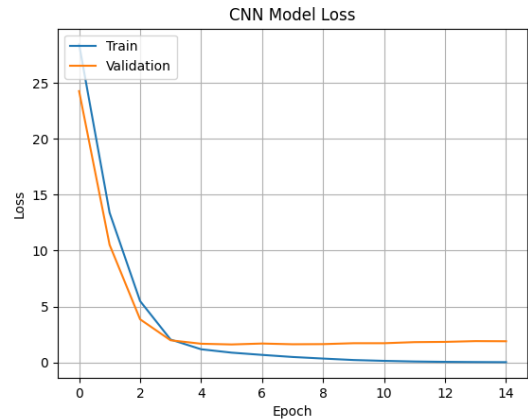
Obr. 9.4 - CNN: Ztráta modelu pro experiment č. 1

První trénování modelu probíhalo s jednou konvoluční vrstvou po dobu 15 epoch a velikostí dávky 32.

Průběh validace po prvním testování byl 40 % se ztrátou 200 %



Obr. 9.5 - CNN: Validace modelu pro experiment č. 2

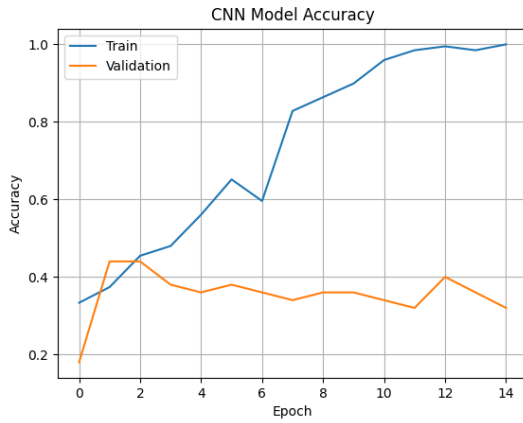


Obr. 9.6 - CNN: Ztráta modelu pro experiment č. 2

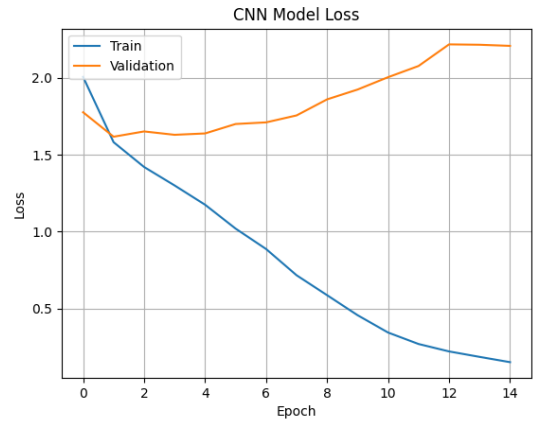
Pro druhý experiment jsem zvýšil počet filtrů v konvoluční síti z 32 na 64. Výsledkem druhého průběhu trénování je validace 42 % a ztráta 190 %.

Ve třetím experimentu byla přidána Max poolingová vrstva (*MaxPooling2D*) za první konvoluční vrstvu, která provádí převzorkování na menší velikost (downsampling). Vrstva je nastavena s parametry:

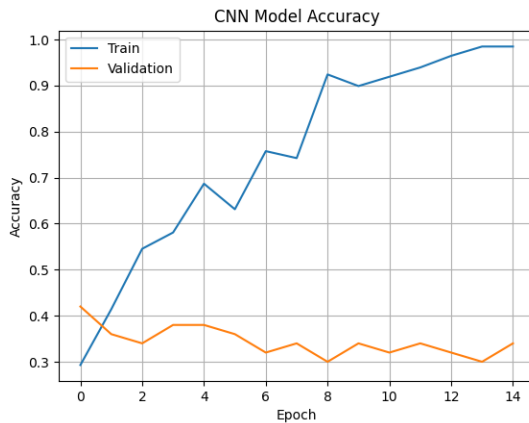
- $pool\_size=(3,2)$  – Velikost jádra, kterým se bude posouvat po vstupu pro nalezení maxima;
- $strides=(1,1)$  – O kolik pixelů se bude okno posouvat při zpracování zpracování vstupu. V tomto případě o 1 pixel v obou rozměrech. Přidána byla i druhá konvoluční vrstva s 128 filtry a stejnou velikostí jádra filtru jako u předchozí vrstvy. Za druhou konvoluční vrstvu byla také přidána Max pooling vrstva. Dále jsem přidal techniky pro snižování rychlosti učení pro optimalizaci sítě, *LearningRateScheduler (0,001)* a *ReduceLROnPlateau*. Výsledkem třetího experimentu je přesnost validace 34 % se ztrátou 200 %.



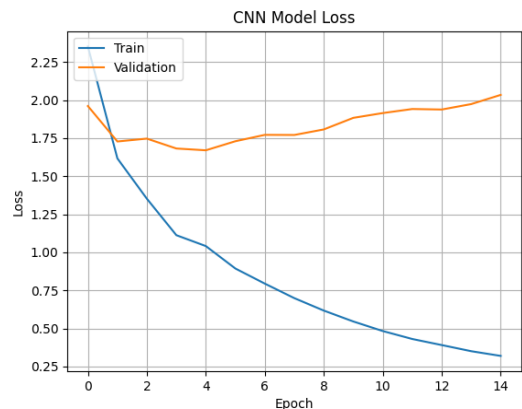
Obr. 9.7 - CNN: Validace modelu pro experiment č. 4



Obr. 9.8 - CNN: Ztráta modelu pro experiment č. 4



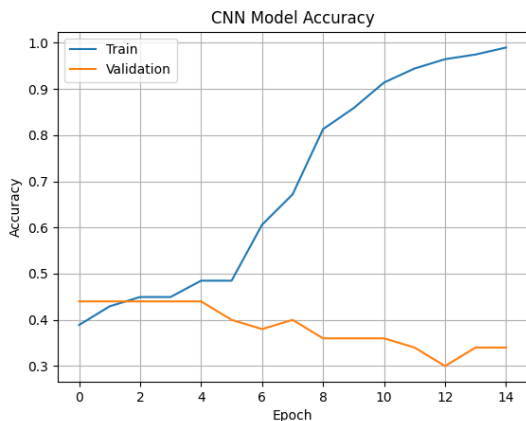
Obr. 9.9 - CNN: Validace modelu pro experiment č. 3



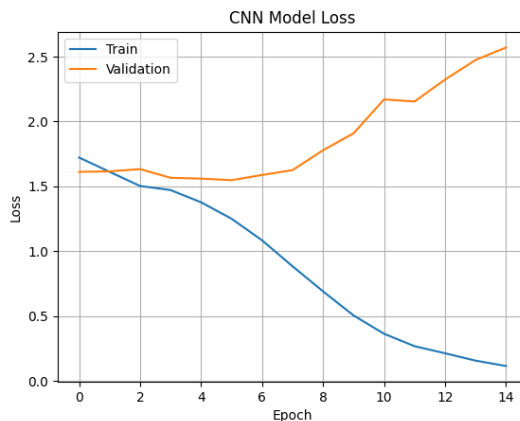
Obr. 9.10 - CNN: Ztráta modelu pro experiment č. 3

Pro čtvrtý experiment byla přidána další, pořadí již 3. konvoluční vrstva s 256 filtry. Stále se používá aktivační funkce ReLU. Výsledná validace pro tento průběh byla 32 % a ztráta 226 %.

V pátém experimentu byla za poslední přidanou konvoluční vrstvu přidána max pooling vrstva a následně čtvrtá konvoluční vrstva s 128 filtry. Výsledkem tohoto experimentu je validace 34 % a ztráta 268 %.

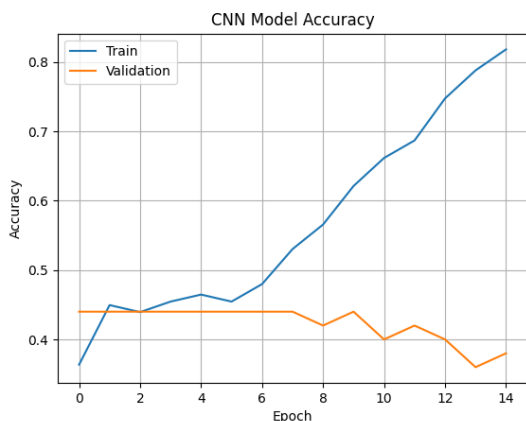


Obr. 9.11 - CNN: Validace modelu pro experiment č. 5

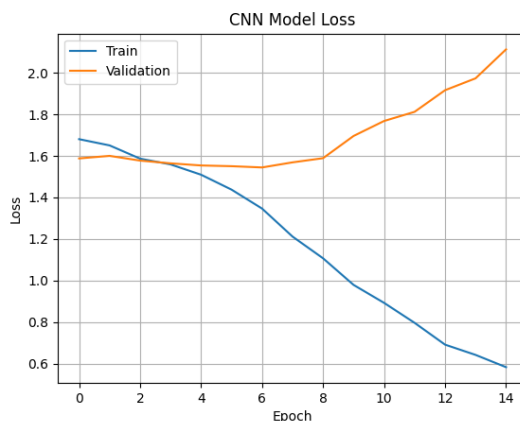


Obr. 9.12 - CNN: Ztráta modelu pro experiment č. 5

Pro šestý experiment byla přidána plně propojená vrstva s 64 neurony (*Dense*) a aktivační funkcí ReLU. Dále byla přidána vrstva *Dropout(0.5)*, tato vrstva slouží jako technika regularizace pro prevenci nadměrného učení. Výsledná validace byla 38 % a ztráta 220 %.

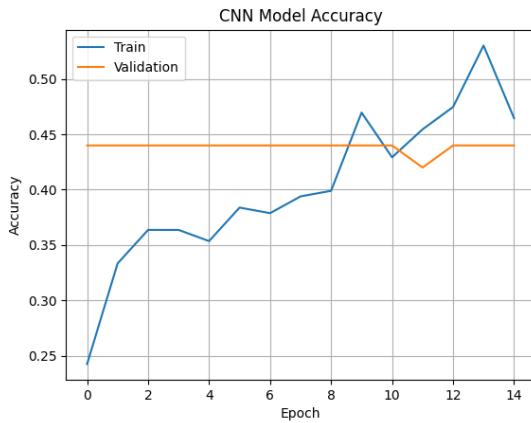


Obr. 9.13 - CNN: Validace modelu pro experiment č. 6

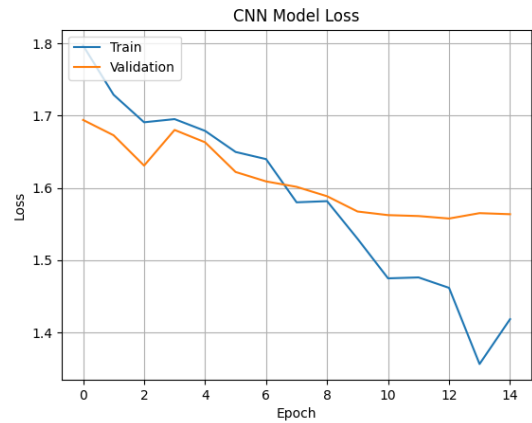


Obr. 9.14 - CNN: Ztráta modelu pro experiment č. 6

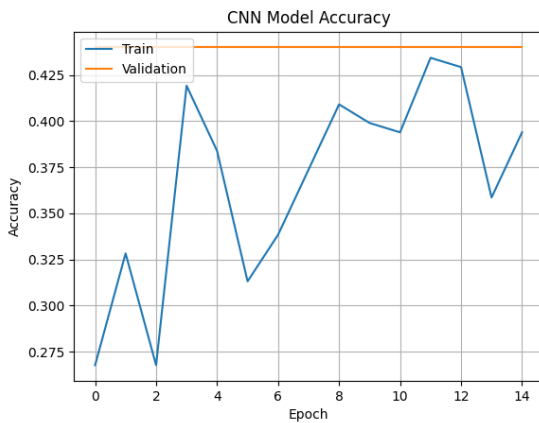
Pro sedmý experiment byla přidána druhá plně propojená vrstva s 64 neurony a *Dropout(0.3)*. Výsledkem tohoto experimentu je validační množina 44 % se ztrátou 157 %.



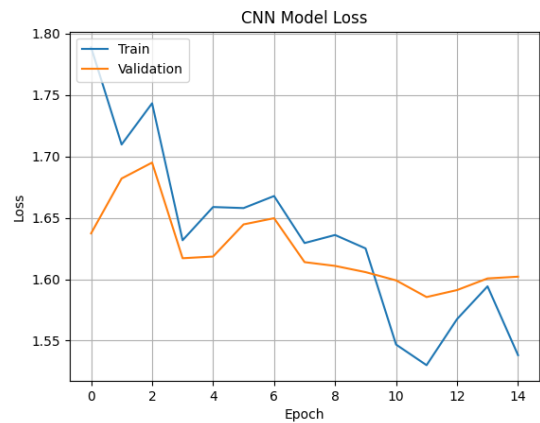
Obr. 9.15 - CNN: Validace modelu pro experiment č. 7



Obr. 9.16 - CNN: Ztráta modelu pro experiment č. 7



Obr. 9.17 - CNN: Validace modelu pro experiment č. 8



Obr. 9.18 - CNN: Ztráta modelu pro experiment č. 8

V osmém experimentu jsem přidal třetí plně propojenou vrstvu s 64 neurony a vrstvu *Dropout(0.1)*. Výsledkem tohoto experimentu je validace 44 % se ztrátou 160 %

## 10 DISKUZE

V praktické části jsem se zaměřil na klasifikaci dat pacientů trpící parézou lícního nervu za použití neuronových sítí. Data poskytnutá oddělením ORL Fakultní nemocnice Královské Vinohrady obsahovala soubor 260 pacientů, kteří byli v průběhu rehabilitace snímáni 3D kamerou. Tento datový soubor mi poskytl základ pro provedení experimentů pomocí neuronových sítí typu LSTM, GRU a CNN. Cílem těchto experimentů bylo objektivní vyhodnocení výkonu daných neuronových sítí na předem neviděných datech.

Jako první jsem si vybral síť LSTM, protože tyto sítě jsou efektivním nástrojem pro studium a modelování sekvenčních dat a mají různorodé využití, hlavně tedy pro modelování sekvenčních dat. Tabulka všech provedených experimentů s dosaženými výsledky pro validační množinu a ztrátu:

Tab. 10.1 - Tabulka výsledků trénování pro LSTM

Experiment	Validační přesnost	Ztráta
1	46,8 %	140,9 %
2	47,5 %	152,1 %
3	46,9 %	145,6 %
4	54,3 %	122,1 %
5	54,3 %	118,7 %
6	61,0 %	106,3 %
7	56,2 %	26,4 %
8	67,7 %	22,7 %

Jako druhou jsem zvolil síť GRU, která vychází z první použité neuronové sítě a je jednodušší. Vnitřní struktura GRU je v porovnání s LSTM jednodušší a snadněji se trénuje, jelikož je do ní zapojeno méně výpočtů.

Tab. 10.2 - Tabulka výsledků trénování pro GRU

Experiment	Validační přesnost	Ztráta
1	52 %	31,1 %
2	49 %	31,5 %
3	57,6 %	26,4 %
4	51,6 %	29,5 %
5	52,6 %	28,2 %
6	54,3 %	27,7 %
7	57,2 %	27,2 %
8	64,4 %	23,4 %

Jako třetí síť jsem zvolil CNN, která se od LSTM a GRU liší architekturou a principem fungování. CNN se běžně používá pro zpracování obrazových dat, ale lze ji využít

i pro analýzu časových řad. Na rozdíl od LSTM a GRU, které pracují s celou sekvencí dat najednou, CNN zpracovává data po oknech a využívá konvoluční operace k extrakci lokálních vzorců.

Tab. 10.3 - Tabulka výsledků trénování pro CNN

Experiment	Validační přesnost	Ztráta
1	40 %	200 %
2	42 %	190 %
3	34 %	200 %
4	32 %	226 %
5	34 %	268 %
6	38 %	220 %
7	44 %	157 %
8	44 %	160 %

Nyní bude následovat tabulka s vyhodnocením jednotlivých sítí, která shrnuje výsledky experimentů se třemi neuronovými sítěmi (LSTM, GRU a CNN) pro klasifikaci dat pacientů po operaci rovnovážného ústrojí.

Tab. 10.4 - Tabulka výsledků pro všechny sítě

Experiment	Validace	Ztráta	Validace	Ztráta	Validace	Ztráta
	LSTM	LSTM	GRU	GRU	CNN	CNN
1	46,8 %	140,9 %	52 %	31,1 %	40 %	200 %
2	47,5 %	152,1 %	49 %	31,5 %	42 %	190 %
3	46,9 %	145,6 %	57,6 %	26,4 %	34 %	200 %
4	54,3 %	122,1 %	51,6 %	29,5 %	32 %	226 %
5	54,3 %	118,7 %	52,6 %	28,2 %	34 %	268 %
6	61,0 %	106,3 %	54,3 %	27,7 %	38 %	220 %
7	56,2 %	26,4 %	57,2 %	27,2 %	44 %	157 %
8	67,7 %	22,7 %	64,4 %	23,4 %	44 %	160 %
Průměrná validace	54,34 %		55 %		39 %	

Analýza validační přesnosti:

- Síť GRU dosáhla nejvyšší průměrné přesnosti 55 %, čímž překonala síť LSTM (55,34 %) a CNN (39 %). To naznačuje, že síť GRU byla nejefektivnější v klasifikaci dat pacientů;

- Síť LSTM dosáhla druhé nejvyšší validační přesnosti 54,34 % což je srovnatelný výsledek s GRU. I když LSTM nedosáhla nejvyšší průměrné přesnosti, stále se ukázala jako efektivní nástroj pro klasifikaci dat;
- Síť CNN dosáhla nejnižší průměrné validační přesnosti 39 %. To naznačuje, že tato síť nebyla tak efektivní pro klasifikaci dat v tomto specifickém úkolu.

Analýza ztráty:

- Sítě LSTM a GRU dosahovaly v průměru nižší ztráty než síť CNN;
- Ztráta všech sítí se s postupem experimentů snižovala.

Výsledky této analýzy naznačují, že síť GRU a LSTM jsou v případě provedených experimentů vhodnější pro klasifikaci dat pacientů po operaci rovnovážného ústrojí než síť CNN. Obě sítě dosáhly vyšší validační přesnosti a zároveň dosahovaly nižší ztráty.

Je však důležité poznamenat, že se jedná o výsledky z relativně malého počtu experimentů. Pro definitivní závěry by bylo nutné provést více experimentů s větším souborem dat a s různými parametry sítí.

Představené výsledky klasifikace nejsou optimální. Existuje prostor pro zlepšení přesnosti a spolehlivosti klasifikace. Klíčové faktory, které mohou ovlivnit výsledky klasifikace, jsou:

- Kvalita dat. Je pravděpodobné, že vhodné předzpracování dat by mohlo vést ke zlepšení přesnosti klasifikace, např. zařadit do datasetu klasifikaci i rychlosti – první derivace atp.;
- Množství dat. Je pravděpodobné, že na větším datasetu by se dané architektury lépe natrénovaly a vykazovaly tak vyšší přesnost;
- Vyváženost dat. Je pravděpodobné, že vyvážený dataset by rovněž umožnil efektivnější trénování sítě.

Dalším faktorem, který je nutné zvážit je výpočetní náročnost sítí. Síť LSTM a GRU jsou obecně méně náročné na výpočetní zdroje než síť CNN. To může být dalším faktorem v praxi, kde je potřeba klasifikovat data v reálném čase.

## ZÁVĚR

Výsledky této práce naznačují, že neuronové sítě mohou být efektivním nástrojem pro klasifikaci dat pacientů po operaci rovnovážného ústrojí a pro hodnocení změn v jejich pohyblivosti během rehabilitačního procesu. Síť LSTM a GRU přinesly uspokojivé výsledky, kdežto síť CNN v tomto experimentu nedosahovala očekávané přesnosti klasifikace.

V kontextu této diplomové práce je důležité zmínit koncept přibližné klasifikace. Při hodnocení výkonu neuronových sítí pro klasifikaci dat pacientů po operaci rovnovážného ústrojí můžeme zohlednit i přibližně správné klasifikace. To znamená, že síť klasifikuje data do sousední třídy (třídy nacházející se o jednu pozici výše nebo níže v hierarchii tříd) oproti přesné klasifikaci do správné třídy.

Důležitým aspektem této práce je, že jsem mohl využít reálná data pacientů, která byla poskytnuta Klinikou ORL FNKV Praha. To dává výsledkům studie větší validitu a využitelnost v praxi.

Jedním limitem této práce je relativně malý počet experimentů, které byly pro každou síť provedeny. V budoucnu by bylo vhodné provést více experimentů s různými parametry sítí a s větším souborem dat.

Dalším limitem je, že tato studie neanalyzovala vliv specifických rehabilitačních technik na změny v pohyblivosti pacientů. V budoucnu by bylo možné provést studii, která by porovnála různé rehabilitační programy a jejich vliv na zotavování pacientů.

Výsledky této práce mohou mít několik důležitých implikací pro praxi:

- **Objektivní evaluace rehabilitačního procesu:** Neuronové sítě mohou být využity pro objektivní evaluaci rehabilitačního procesu pacientů po operaci rovnovážného ústrojí. To může lékařům a fyzioterapeutům pomoci lépe sledovat pokrok pacientů a přizpůsobovat rehabilitační programy individuálním potřebám pacientů;
- **Optimalizace rehabilitačních programů:** Výsledky této práce mohou být použity k optimalizaci rehabilitačních programů pacientů po operaci rovnovážného ústrojí. Například mohou být identifikovány nejefektivnější rehabilitační techniky a tyto techniky mohou být následně integrovány do rehabilitačních programů;
- **Vývoj nových diagnostických nástrojů:** Neuronové sítě mohou být použity k vývoji nových diagnostických nástrojů pro pacienty po operaci rovnovážného ústrojí. Tyto nástroje by mohly lékařům a fyzioterapeutům pomoci s včasnou diagnózou a plánováním rehabilitace.

Jako laik jsem neměl povědomí, že může nastat poškození lícního nervu a jeho paréza po operaci rovnovážného ústrojí, ale i u onemocnění neurologických postihujících lícní nerv. Další jsou skupinou pacienti po centrálních mozkových příhodách, kde je opět poškození lícního nervu. S využitím neuronových sítí a snímání obličeje při rehabilitaci lze cíleně procvičovat postiženou oblast, a tím zkrátit dobu léčby o několik týdnů až měsíců, a následně tak zlepšit kvalitu života nemocného pacienta.

Přínos do praxe by mohl být takový, že by lékařům a fyzioterapeutům snímání obličeje umožnilo rozpoznat problém, která oblast lícního nervu je poškozena. V současné době se využívá EMG na zjištění síly inervovaného svalu lícním nervem. Toto vyšetření je bolestivé, a ne zcela přesné. V budoucnu by se mohlo aplikovat jako vedlejší vyšetření k snímání obličeje.

## POUŽITÁ LITERATURA

- ALI, Zulfiqar a SBala BHASKAR, 2016. Basic statistical tools in research and data analysis. *Indian Journal of Anaesthesia* [online]. **60**(9), 8 [cit. 2024-01-29]. ISSN 0019-5049. Dostupné z: doi:10.4103/0019-5049.190623
- ALICANDRI-CIUFELLI, Matteo, Alessia PICCININI, Alberto GRAMMATICA, et al., 2013. A step backward: The 'Rough' facial nerve grading system. *Journal of Cranio-Maxillofacial Surgery* [online]. **41**(7), e175-e179 [cit. 2024-05-01]. ISSN 10105182. Dostupné z: doi:10.1016/j.jcms.2012.11.047
- ALLOGHANI, Mohamed, Dhiya AL-JUMEILY, Jamila MUSTAFINA, Abir HUSSAIN a Ahmed J. ALJAAF, 2020. A Systematic Review on Supervised and Unsupervised Machine Learning Algorithms for Data Science. In: BERRY, Michael W., Azlinah MOHAMED a Bee Wah YAP, ed. *Supervised and Unsupervised Learning for Data Science* [online]. Cham: Springer International Publishing, 2020, s. 3-21 [cit. 2024-05-02]. Unsupervised and Semi-Supervised Learning. ISBN 978-3-030-22474-5. Dostupné z: doi:10.1007/978-3-030-22475-2\_1
- AMBLER, Zdeněk, Pavel POTUŽNÍK a Jiří POLÍVKA, 2023. *Základy neurologie*. Osmé, aktualizované a doplněné vydání. Praha: Galén. ISBN 978-80-7492-654-9.
- ANSH, David, 2021. Single Layer Perceptron and Activation Function. *Medium* [online]. [cit. 2024-05-10]. Dostupné z: <https://medium.com/codex/single-layer-perceptron-and-activation-function-b6b74b4aae66>
- DING, Ying, Ronald ROUSSEAU a Dietmar WOLFRAM, ed., 2014. *Measuring Scholarly Impact* [online]. Cham: Springer International Publishing [cit. 2024-04-25]. ISBN 978-3-319-10376-1. Dostupné z: doi:10.1007/978-3-319-10377-8
- DOLEŽEL, Petr, 2016. *Úvod do umělých neuronových sítí pro studenty technických vysokých škol* [online]. Univerzita Pardubice, 78 s. [cit. 2024-05-02]. ISBN 978-80-7560-022-6. Dostupné z: <https://eshop.upce.cz/epub/9004940/>
- DRUGA, Rastislav, Miloš GRIM a Petr DUBOVÝ, 2011. *Anatomie centrálního nervového systému*. Praha: Galén. ISBN 978-80-7262-706-6.
- ELMAZ, Furkan, Reinout EYCKERMAN, Wim CASTEELS, Steven LATRÉ a Peter HELLINCKX, 2021. CNN-LSTM architecture for predictive indoor temperature modeling. *Building and Environment* [online]. **206**(1), 9 [cit. 2024-04-01]. ISSN 03601323. Dostupné z: doi:10.1016/j.buildenv.2021.108327
- ELSHEIKH, Ahmed, Soumaya YACOUT a Mohamed-Salah OUALI, 2019. Bidirectional handshaking LSTM for remaining useful life prediction. *Neurocomputing* [online]. **323**(1), 148-156 [cit. 2024-04-01]. ISSN 09252312. Dostupné z: doi:10.1016/j.neucom.2018.09.076
- FIFE, Terry D., 2010. Overview of anatomy and physiology of the vestibular system. In: *Vertigo and Imbalance: Clinical Neurophysiology of the Vestibular System* [online]. Elsevier, s. 5-17 [cit. 2024-05-01]. Handbook of Clinical Neurophysiology. ISBN 9780444529121. Dostupné z: doi:10.1016/S1567-4231(10)09002-7
- HAND, DJ, 1992. Statistical methods in diagnosis. *Statistical Methods in Medical Research* [online]. **1**(1), 49-67 [cit. 2024-01-29]. ISSN 0962-2802. Dostupné z: doi:10.1177/096228029200100104

- HASTIE, Trevor, Robert TIBSHIRANI a Jerome FRIEDMAN, 2009. *The Elements of Statistical Learning* [online]. 2. New York, NY: Springer New York [cit. 2024-02-04]. Springer Series in Statistics. ISBN 978-0-387-84857-0. Dostupné z: doi:10.1007/978-0-387-84858-7
- CHARBUTY, Bahzad a Adnan ABDULAZEEZ, 2021. Classification Based on Decision Tree Algorithm for Machine Learning. *Journal of Applied Science and Technology Trends* [online]. 2021-03-30, **2**(01), 20-28 [cit. 2024-02-17]. ISSN 2708-0757. Dostupné z: doi:10.38094/jastt20165
- Illustrating the random forest algorithm in TikZ, 2019. In: *Stack Exchange* [online]. 2019 [cit. 2024-05-02]. Dostupné z: <https://tex.stackexchange.com/questions/503883/illustrating-the-random-forest-algorithm-in-tikz>
- INTWALA, Aditya M. a Atul MAGIKAR, 2016. A review on process of 3D Model Reconstruction. In: *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)* [online]. Chennai, India: IEEE, s. 2851-2855 [cit. 2024-04-08]. ISBN 978-1-4673-9939-5. Dostupné z: doi:10.1109/ICEEOT.2016.7755218
- JANUROVÁ, Kateřina, 2011. *Regresní analýza, 13. cvičení* [online]. [cit. 2024-05-02]. Dostupné z: [https://homel.vsb.cz/~jan939/STA/cviceni/13\\_Regresni\\_analyza.pdf](https://homel.vsb.cz/~jan939/STA/cviceni/13_Regresni_analyza.pdf)
- JIANG, Tammy, Jaimie L. GRADUS a Anthony J. ROSELLINI, 2020. Supervised Machine Learning: A Brief Primer. *Behavior Therapy* [online]. **51**(5), 675-687 [cit. 2024-05-03]. ISSN 00057894. Dostupné z: doi:10.1016/j.beth.2020.05.002
- KONUKOGLU, Ender a Ben GLOCKER, 2020. Random forests in medical image computing. In: *Handbook of Medical Image Computing and Computer Assisted Intervention* [online]. Elsevier, s. 457-480 [cit. 2024-02-23]. ISBN 9780128161760. Dostupné z: doi:10.1016/B978-0-12-816176-0.00024-7
- KURISCAK, Eduard, Petr MARSALEK, Julius STROFFEK a Peter G. TOTH, 2015. Biological context of Hebb learning in artificial neural networks, a review. *Neurocomputing* [online]. **152**, 27-35 [cit. 2024-05-02]. ISSN 09252312. Dostupné z: doi:10.1016/j.neucom.2014.11.022
- KUSYN, Michal, 2020. Obrna lícního nervu (paréza nervi facialis, Bellova obrna). *AkuFyzio* [online]. [cit. 2024-05-01]. Dostupné z: <https://aku-fyzio.cz/obrna-licniho-nervu/>
- LI, Chuang, Guojie LI, Keyou WANG a Bei HAN, 2022. A multi-energy load forecasting method based on parallel architecture CNN-GRU and transfer learning for data deficient integrated energy systems. *Energy* [online]. **259** [cit. 2024-05-03]. ISSN 03605442. Dostupné z: doi:10.1016/j.energy.2022.124967
- LIM, Huey Woan, Yuan Wen HAU, Chiao Wen LIM a Mohd Afzan OTHMAN, 2016. Artificial intelligence classification methods of atrial fibrillation with implementation technology. *Computer Assisted Surgery* [online]. 2016-10-25, **21**(sup1), 154-161 [cit. 2024-02-13]. ISSN 2469-9322. Dostupné z: doi:10.1080/24699322.2016.1240303
- LYNCH, Patrick J., 2006. Cranial nerve VII. In: *Wikimedia Commons* [online]. [cit. 2024-05-02]. Dostupné z: [https://commons.wikimedia.org/wiki/File:Cranial\\_nerve\\_VII.svg#filelinks](https://commons.wikimedia.org/wiki/File:Cranial_nerve_VII.svg#filelinks)

- MICHIE, D., 1995. *Machine learning, neural and statistical classification*. New York: Ellis Horwood. ISBN 9780131063600.
- NÁRODNÍ ZDRAVOTNICKÝ INFORMAČNÍ PORTÁL, neuvedeno. *Ucho, sluch a rovnováha* [online]. [cit. 2024-04-25]. ISSN 2695-0340. Dostupné z: <https://www.nzip.cz/clanek/1507-ucho-sluch-rovnovaha>
- NOSOUHAIN, Shiva, Fereshteh NOSOUHAIN a Abbas Kazemi KHOSHOUEI, 2021. *A Review of Recurrent Neural Network Architecture for Sequence Learning: Comparison between LSTM and GRU* [online]. 7 [cit. 2024-05-03]. Dostupné z: [doi:https://doi.org/10.20944/preprints202107.0252.v1](https://doi.org/10.20944/preprints202107.0252.v1)
- OLAH, Christopher, 2015. *Understanding LSTM Networks* [online]. [cit. 2024-05-03]. Dostupné z: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/#fnref1>
- O'SHEA, Keiron a Ryan NASH, 2015. *An Introduction to Convolutional Neural Networks* [online]. 10 [cit. 2024-04-03]. Dostupné z: [doi:https://doi.org/10.48550/arXiv.1511.08458](https://doi.org/10.48550/arXiv.1511.08458)
- OTTAIANO, Ana Carolina, Gustavo Dalul GOMEZ a Tomás de Andrade Lourenção FREDDI, 2023. The Facial Nerve: Anatomy and Pathology. *Seminars in Ultrasound, CT and MRI* [online]. **44**(2), 71-80 [cit. 2024-05-01]. ISSN 08872171. Dostupné z: [doi:10.1053/j.sult.2022.11.005](https://doi.org/10.1053/j.sult.2022.11.005)
- PATEL, Harsh H. a Purvi PRAJAPATI, 2018. Study and Analysis of Decision Tree Based Classification Algorithms. *International Journal of Computer Sciences and Engineering* [online]. 2018-10-31, **6**(10), 74-78 [cit. 2024-02-17]. ISSN 23472693. Dostupné z: [doi:10.26438/ijcse/v6i10.7478](https://doi.org/10.26438/ijcse/v6i10.7478)
- PRASAD, Sashank a Steven L. GALETTA, 2011. Anatomy and physiology of the afferent visual system. In: *Neuro-ophthalmology* [online]. Elsevier, s. 3-19 [cit. 2024-05-01]. Handbook of Clinical Neurology. ISBN 9780444529039. Dostupné z: [doi:10.1016/B978-0-444-52903-9.00007-8](https://doi.org/10.1016/B978-0-444-52903-9.00007-8)
- RAMESH, AN, C KAMBHAMPATI, JRT MONSON a PJ DREW, 2004. Artificial intelligence in medicine. *Annals of The Royal College of Surgeons of England* [online]. 2004-09-01, **86**(5), 334-338 [cit. 2024-05-03]. ISSN 00358843. Dostupné z: [doi:10.1308/147870804290](https://doi.org/10.1308/147870804290)
- RIGATTI, Steven J., 2017. Random Forest. *Journal of Insurance Medicine* [online]. 2017-01-01, **47**(1), 31-39 [cit. 2024-02-23]. ISSN 0743-6661. Dostupné z: [doi:10.17849/in-sm-47-01-31-39.1](https://doi.org/10.17849/in-sm-47-01-31-39.1)
- ROBINSON, Mara Wernick a Jennifer BAIUNGO, 2018. Facial Rehabilitation. *Otolaryngologic Clinics of North America* [online]. **51**(6), 1151-1167 [cit. 2023-12-08]. ISSN 00306665. Dostupné z: [doi:10.1016/j.otc.2018.07.011](https://doi.org/10.1016/j.otc.2018.07.011)
- SAK, Haşim, Andrew SENIOR a Françoise BEAUFAYS, 2014. *Long Short-Term Memory Based Recurrent Neural Network Architectures for Large Vocabulary Speech Recognition* [online]. [cit. 2024-05-03]. Dostupné z: [doi:https://doi.org/10.48550/arXiv.1402.1128](https://doi.org/10.48550/arXiv.1402.1128)
- SANTOS, Carla F., Jorge BELINHA, Fernanda GENTIL, Marco PARENTE, Bruno AREIAS a Renato Natal JORGE, 2017. Biomechanical Study of the Vestibular System of the Inner Ear Using a Numerical Method. *Procedia IUTAM* [online]. **24**, 30-37 [cit. 2024-04-25]. ISSN 22109838. Dostupné z: [doi:10.1016/j.piutam.2017.08.040](https://doi.org/10.1016/j.piutam.2017.08.040)

- SARICA, Alessia, Antonio CERASA a Aldo QUATTRONE, 2017. Random Forest Algorithm for the Classification of Neuroimaging Data in Alzheimer's Disease: A Systematic Review. *Frontiers in Aging Neuroscience* [online]. 2017-10-06, **9**(1), 12 [cit. 2024-02-23]. ISSN 1663-4365. Dostupné z: doi:10.3389/fnagi.2017.00329
- SCHMIDT, Robin M., 2019. *Recurrent Neural Networks (RNNs): A gentle Introduction and Overview* [online]. 16 [cit. 2024-05-03]. Dostupné z: doi:https://doi.org/10.48550/arXiv.1912.05911
- THIERFELDER, Susanne, 2022. *Create your own depth measuring tool with MediaPipe FaceMesh in Javascript* [online]. [cit. 2024-04-23]. Dostupné z: https://medium.com/@susanne.thierfelder/create-your-own-depth-measuring-tool-with-mediapipe-facemesh-in-javascript-ae90abae2362
- VOKURKA, Martin a Jan HUGO, 2023. *Praktický slovník medicíny*. 12. aktualizované vydání. Praha: Maxdorf. ISBN 978-80-7345-770-9.
- WANG, Jane-Ling, Jeng-Min CHIOU a Hans-Georg MÜLLER, 2016. Functional Data Analysis. *Annual Review of Statistics and Its Application* [online]. 2016-06-01, **3**(1), 257-295 [cit. 2024-01-29]. ISSN 2326-8298. Dostupné z: doi:10.1146/annurev-statistics-041715-033624
- WONG, Brian J.-F, Michelle G. ARNOLD a Jacob O. BOECKMANN, ed., 2016. *Facial Plastic and Reconstructive Surgery* [online]. Cham: Springer International Publishing [cit. 2024-05-01]. ISBN 978-3-319-18034-2. Dostupné z: doi:10.1007/978-3-319-18035-9
- YUAN, Ann a Andrey VAKUNOV, 2020. *Face and hand tracking in the browser with mediapipe and Tensorflow.js* [online]. 9.3. 2020 [cit. 2024-04-24]. Dostupné z: https://blog.tensorflow.org/2020/03/face-and-hand-tracking-in-browser-with-mediapipe-and-tensorflowjs.html
- YUNIARTI, Anny a Nanik SUCIATI, 2019. A Review of Deep Learning Techniques for 3D Reconstruction of 2D Images. In: *2019 12th International Conference on Information & Communication Technology and System (ICTS)* [online]. Surabaya, Indonesia: IEEE, s. 327-331 [cit. 2024-04-08]. ISBN 978-1-7281-2133-8. Dostupné z: doi:10.1109/ICTS.2019.8850991
- ZABOR, Emily C., Chandana A. REDDY, Rahul D. TENDULKAR a Sujata PATIL, 2022. Logistic Regression in Clinical Studies. *International Journal of Radiation Oncology\*Biophysics* [online]. **112**(2), 271-277 [cit. 2024-02-06]. ISSN 03603016. Dostupné z: doi:10.1016/j.ijrobp.2021.08.007