



ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

NEUMANN JÁNOS
INFORMATIKAI KAR



SZAKDOLGOZAT

OE-NIK Hallgató neve:
2023 Hallgató törzskönyvi száma:

Megyes István
T/006303/FI12904/N



ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

**NEUMANN JÁNOS
INFORMATIKAI KAR**



Elektroenkefalográfia jelek feldolgozása Gépi tanulással

OE-NIK Hallgató neve:
2023 Hallgató törzskönyvi száma:

Megyes István
T/006303/FI12904/N

Óbudai Egyetem
Neumann János Informatikai Kar
Biomatika és Alkalmazott Mesterséges Intelligencia Intézet

SZAKDOLGOZAT FELADATLAP

Hallgató neve: **Megyes István**
Törzskönyvi száma: T/006303/FI12904/N
Neptun kódja: L38TRN

A dolgozat címe:

Elektroenkefalográfia jelek feldolgozása Gépi tanulással
Processing electroencephalography signals using Machine learning

Intézményi konzulensek: Nemes Gyula Ádám
Külső konzulens:

Beadási határidő: 2023. december 15.

A záróvizsga tárgyai: Számítógép architektúrák
Mesterséges intelligencia specializáció

A feladat

Az EEG jelek feldolgozása hagyományosan kézzel készített szűrők segítségével történik, ez azonban időigényes és nagy gyakorlati háttérrel igénylő folyamat.

Az utóbbi években sikerrel alkalmaztak különböző gépi tanulás módszereket agyi jelek feldolgozására. Kellő mennyiségű adat birtokában a neurális hálók különösen alkalmasak hasonló komplexitású forrásból származó minták felismerésére.

A jelölt feladata egy Event Related Potential (ERP) paradigma kidolgozása, amely segítségével gondolatokkal vezérelhetjük a számítógépet.

Ezt követően a feladathoz szükséges adatvagyon előállítása, értelmezése, előfeldolgozása, tisztítása.

Majd az adatok értelmezésre szolgáló neurális háló feltanítása, végül így egy egyszerű Agy-Számítógép interfész létrehozása.

A dolgozatnak tartalmaznia kell:

- szakirodalom felmérését,
- ERP paradigma kidolgozását,
- adattisztítási folyamatok kidolgozását,
- Neurális háló architektúrák összehasonlító elemzése,
- gépi tanulás modell feltanítását,
- az eredmények kiértékelését.



.....
Dr. Eigner György
intézetigazgató

A szakdolgozat elévülésének határideje: **2025. december 15.**
(OE TVSz 55.§ szerint)

A dolgozatot beadásra alkalmasnak tartom:

.....
külső konzulens

.....
intézményi konzulens



HALLGATÓI NYILATKOZAT

Alulírott hallgató kijelentem, hogy a szakdolgozat saját munkám eredménye, a felhasznált szakirodalmat és eszközöket azonosíthatóan közöltem. Az elkészült szakdolgozatomba található eredményeket az egyetem és a feladatot kiíró intézmény saját céljára térítés nélkül felhasználhatja.

Budapest, 2023.06.02.....

hallgató aláírása



KONZULTÁCIÓS NAPLÓ

Hallgató neve:

Neptun Kód:

Tagozat:

Megyes István.....

L38TRN.....

nappali.....

Telefon: ..

Levelezési cím (pl.: lakcím):

+36305062280 1204, Bp. Török Flóris utca 153.....

Szakdolgozat / Diplomamunka¹ címe magyarul:

Elektroencephalography jelek feldolgozása Gépi tanulással.....

Szakdolgozat / Diplomamunka² címe angolul:

Processing electroencephalography signals using Machine learning.....

Intézményi konzulens:

Külső konzulens:

Nemes Gyula Ádám.....

.....

Kérjük, hogy az adatokat nyomtatott nagy betűkkel írja!

Alk.	Dátum	Tartalom	Aláírás
1.	2023.02.28	Szakdolgozat felépítéséhez tanácsadás.	
2.	2023.03.24	Feladatlap áttekintése, feladatok pontosabb specifikációja.	
3.	2023.04.17	Szakdolgozathoz szükséges mérés, program tesztelés	
4.	2023.05.15	Leadási határidő, ahhoz szükséges dokumentumok, prezentáció szerkezeti követelményeinek megbeszélése.	

A Konzultációs naplót összesen 4 alkalommal, az egyes konzultációk alkalmával kell láttamoztatni bármelyik konzulenssel.

A hallgató a Szakdolgozat / Szakdolgozat I. / Szakdolgozat II. / Projektlabor 1 / Projektlabor 2 / Projektlabor 3 / Záródolgozati projekt / Diplomamunka I / Diplomamunka II / Diplomamunka III / Diplomamunka IV³ tantárgy követelményét teljesítette, beszámolóra / védésre⁴ bocsátható.

Budapest, 2023.05.15.....

Intézményi konzulens

¹ Megfelelő aláhúzendő!

² Megfelelő aláhúzendő!

³ Megfelelő aláhúzendő!

⁴ Megfelelő aláhúzendő!



KONZULTÁCIÓS NAPLÓ

Hallgató neve: Neptun Kód: Tagozat:
Megyes István..... L38TRN..... nappali.....

Telefon: Levelezési cím (pl.: lakcím):
+36305062280 1204, Bp. Török Flóris utca 153.....

Szakdolgozat / Diplomamunka¹ címe magyarul:
Elektroencephalography jelek feldolgozása Gépi tanulással.....

Szakdolgozat / Diplomamunka² címe angolul:
Processing electroencephalography signals using Machine learning.....

Intézményi konzulens: Külső konzulens:
Nemes Gyula Ádám.....

Kérjük, hogy az adatokat nyomtatott nagy betűkkel írja!

Alk.	Dátum	Tartalom	Aláírás
1.	2024.02.21	Félév során elvégzendő lépések meghatározása.	
2.	2024.03.04	Feladat megvalósításának megtervezése	
3.	2024.04.04	Fejlesztési kérdés, komponensek összeillesztése	
4.	2024.05.08	Dolgozat véglegesítése és véleményezése.	

A Konzultációs naplót összesen 4 alkalommal, az egyes konzultációk alkalmával kell láttamoztatni bármelyik konzulenssel.

A hallgató a Szakdolgozat / Szakdolgozat I. / Szakdolgozat II. / Projektlabor 1 / Projektlabor 2 / Projektlabor 3 / Záródolgozati projekt / Diplomamunka I / Diplomamunka II / Diplomamunka III / Diplomamunka IV³ tantárgy követelményét teljesítette, beszámolóra / védésre⁴ bocsátható.

A konzulens által javasolt érdemjegy: 5.....

Budapest, 2024.05.08.....

Intézményi konzulens

¹ Megfelelő aláhúzendó!

² Megfelelő aláhúzendó!

³ Megfelelő aláhúzendó!

⁴ Megfelelő aláhúzendó!

ABSZTRAKT

Az agy-számítógép interfész (BCI) egy olyan speciális érintkezési felület, amely lehetővé teszi a kommunikációt az emberi agy és a számítógép között. Ennek az újfajta megközelítésnek köszönhetően az ember képes lehet a gondolatival kommunikálni. Az elektroenkefalográfia (EEG) felhasználásával lehetőségünk van BCI rendszerek széles körű megvalósítására. Az EEG értékes betekintést nyújt az agyműködésbe, segíti a különböző neurológiai állapotok tanulmányozását, valamint az időben részletes tájékoztatásokat szolgáltat az agyi aktivitás változásairól. Az EEG jelekben bizonyos ingerek kiváltásakor az agyban megtalálható lesz egy P300 nevű hullám, ami fontos szerepet játszik a figyelem irányításába., Ezért jelentős diagnosztikai és prognosztikai potenciállal bír az ember-számítógép interakcióban.

Szakedolgozatomban egy grafikus BCI programot készítettem el, amit agyhullámokkal lehet irányítani. Ezeket a jeleket, egy g.tec Unicorn Hybrid Black (Unicorn) nevű eszköz segítségével gyűjtöttem, mivel a Unicorn képes EEG adatok felvételére. A rendszerem egy integrált alkalmazást képez, amely négy kulcsfontosságú összetevőből áll: egy a kísérlet megjelenítésért felelő grafikus interfész, egy modul amely adatokat gyűjt az Unicorn eszközből, egy az előző két komponensből összegyűjtött adatokat feldolgozó programrész és végül a gépi tanulást megvalósító modellek.

Az alkalmazás alapvető célja az EEG jelek valós idejű feldolgozása és analízise, melyhez az erre a célra fejlesztett EEGNet nevű neurális hálózatot és a lineáris diszkrimináns analízist fogom alkalmazni. Az EEGNet modell a tanulási fázisban 90% feletti pontosságot ért el, ezzel szemléltetve a neurális hálók adatokból történő jellemzőtanulási képességét. Ezzel szemben az LDA modell, tanulási fázisban 80% feletti pontosságot tudott elérni. Azonban valós körülmények között az EEGNet modell teljesítménye 10 és 20% közötti pontosságra csökkent, ami a túlilleszkedésre utal, míg az LDA modell 60 és 70% közötti pontosságot mutatott, bizonyítva nagyobb robusztusságát és jobb általánosító képességét. A feltanított modelleknek köszönhetően, az elkészült alkalmazás képes a képernyőn megjelenített hangszerek EEG-vel vezérelt kiválasztására.

ABSTRACT

The brain-computer interface (BCI) is a specialized interface that allows communication between the human brain and a computer. Thanks to this novel approach, a person may be able to communicate with their thoughts. By using electroencephalography (EEG), we have the potential to implement a wide range of BCI systems. EEG provides valuable insights into brain function, aids in the study of various neurological conditions, and offers detailed, real-time information about changes in brain activity. When certain stimuli are applied, a wave called P300 can be detected in the brain through EEG signals, which plays a crucial role in directing attention. Therefore, it has significant diagnostic and prognostic potential in human-computer interaction.

In my thesis, I developed a graphical BCI program that can be controlled using brainwaves. These signals were collected with a device called the g.tec Unicorn Hybrid Black (Unicorn), which is capable of recording EEG data. My system comprises an integrated application with four key components: a graphical interface responsible for displaying the experiment, a module that collects data from the Unicorn device, a component that processes the data collected by the previous two components, and finally, models that implement machine learning.

The main goal of the application is the real-time processing and analysis of EEG signals, for which I will use a neural network called EEGNet, specifically developed for this purpose, and linear discriminant analysis. The EEGNet model achieved an accuracy of over 90% during the training phase, demonstrating the neural network's ability to learn features from the data. In contrast, the LDA model achieved an accuracy of over 80% during the training phase. However, in real-world conditions, the performance of the EEGNet model dropped to an accuracy between 10% and 20%, indicating overfitting, whereas the LDA model showed an accuracy between 60% and 70%, proving its greater robustness and better generalization capability. Thanks to the trained models, the completed application can select instruments displayed on the screen controlled by EEG.

Tartalomjegyzék

1. Bevezetés	1
2. A probléma megfogalmazása	2
3. Irodalomkutatás	3
3.1. Központi idegrendszer	3
3.1.1. Neuronok	4
3.2. Agy-számítógép interfész	5
3.2.1. A BCI-k alapelvei	6
3.2.2. Invazív és non-invazív módszerek	7
3.3. Elektroenkefalográfia	8
3.3.1. Motoros paradigma	10
3.3.2. Állandó állapotú vizuálisan kiváltott potenciálok	11
3.3.3. Eseményhez kötött potenciál	12
3.3.4. A P300-as jel	13
3.4. Gépi tanulás	15
3.4.1. A gépi tanulás az EEG területén	16
3.4.2. A gépi tanulás a P300 területén	18
3.4.3. EEG adatokat osztályozó algoritmusok	19
3.4.4. EEG adatok feldolgozása neurális hálókkal	20
3.5. Unicorn Hybrid Black	21
3.6. Felhasználói interfész	22
3.7. A megoldási módszer kiválasztása	23
4. Követelmény specifikáció	25
5. Rendszerterv	27
6. Megvalósítás	31
6.1. Fejlesztői környezet	31
6.2. Verziókövetés	31
6.3. Konfiguráció	32
6.4. A kísérlet grafikus interfésze	32
6.5. A kísérlet leírása	34
6.6. Párhuzamos programozás implementálása	34
6.7. Adatok gyűjtése az Unicornnal	35
6.8. Adatok feldolgozása	37
6.9. Modellek tanítása	39

7. Tesztelés	41
8. Eredmények	42
9. Összefoglaló	44
10. Summary	45
11. Hivatkozások	46

Köszönetnyilvánítás

Ezúton szeretném megköszönni konzulensemnek, Nemes Gyula Ádámnak, hogy kitartott mellettem és rendületlen pozitivitása fáklyaként világította az utat mikor úgy éreztem, hogy eltévedtem.

Külön szeretném megköszönni Tóth Viktornak, hogy hatalmas tudásával és páratlan személyével hozzájárult a dolgozatom megszületéséhez.

Végül szeretnék köszönetet mondani a családomnak, különösképpen édesanyámnak, aki mindig bízott bennem és édesapámnak, aki megmutatta, hogy sosem szabad feladni.

1. Bevezetés

Az elmúlt évtizedekben az emberiség számos területen képes volt új megoldásokat találni, régóta fennálló problémákra és alapvetően olyan találmányokkal megajándékozni fajunkat, amivel meg tudjuk könnyíteni saját és egymás mindennapját, életét. Az önvezető autók megalkotása nem csak a közlekedést tette biztonságosabbá, de az emberek életét is megkönnyítette. Az okostelefonok pedig már olyan sokoldalúak és kifinomultak, hogy egyes számítógépeket is túlszárnyalnak. A virtuális és kiterjesztett valóságok létrehozása ugyanakkor lehetővé teszi az emberek számára, hogy új dolgokat fedezzenek fel és új élményeket szerezzenek.

Ezek a technológiai vívmányok nem másnak köszönhetőek, mint az emberi agynak. Láthatjuk tehát, hogy agyunkban rejlő lehetőségek tárháza közel határtalan. Akadnak azonban olyanok közöttünk, akikben ott van a potenciál, hogy a következő technológiai újítást elhozzák, ellenben nem rendelkeznek a szükséges fizikai képességekkel. Nincs lehetőségük rá, mert születésükkor vagy életük folyamán elvesztették a külvilággal való részleges vagy akár teljes kommunikációs képességüket. Legyen szó neurológiai megbetegedésekről vagy esetleg paralízisről, ezeknek az embereknek az elméje dobozba zárt szellemként várja, hogy egyszer kiszabadulhasson.

Az agyunkban található idegsejtek elektromos jelekkel kommunikálnak egymással. Ezeket az elektromos szignálokat többek között az elektroencefalográfia (EEG) segítségével tudjuk vizsgálni. Ezeket a jeleket digitalizálni tudjuk aminek köszönhetően Agy-gép interfészeket (BCI) tudunk létrehozni. A BCI egy olyan speciális, ember által készített „érintkezési felület”, amely lehetővé teszi a kommunikációt az emberi agy és a számítógép között. Ennek az új fajta megközelítésnek köszönhetően az ember képes lehet a gondolatival „kommunikálni” és kiszabadulni a dobozból.

2. A probléma megfogalmazása

A feladatomban az EEG jelek gépi tanulás segítségével történő feldolgozását tűztem ki. Az EEG jelek hagyományosan kézzel készített szűrőkkel kerülnek feldolgozásra, ami időigényes és sok szakmai tapasztalatot igényel. Az elmúlt években azonban sikeresen alkalmaztak különböző gépi tanulás módszereket az agyi jelek feldolgozására, különösen a neurális hálók bizonyultak alkalmasnak hasonlóan összetett minták felismerésére, amennyiben elegendő adat áll rendelkezésre. [1]

Szakdolgozatomban elsősorban áttekintem a releváns szakirodalom állapotát, hogy megismerjem a területen jelenleg elérhető legújabb eredményeket és módszereket. Ezután kiválasztom, hogy melyik paradigmát fogom alkalmazni, amely majd lehetővé teszi, hogy a gondolatokkal vezérelhessük a számítógépet. Az adatok tisztításához kidolgozok egy hatékony folyamatot, amely segít eltávolítani a zajt és a nem kívánt műtermékeket az EEG jelekből.

A következő lépésben összehasonlító elemzést végzek különböző neurális háló architektúrák között, hogy meghatározzam, melyik a legalkalmasabb az EEG jelek felismerésére és feldolgozására. Ezt követően feltanítom a gépi tanulás modellt az összegyűjtött tanító adatok alapján. Végül értékelem a megkapott eredményeket, hogy megállapítsam a modell teljesítményét és hatékonyságát a BCI elkészítéséhez szükséges agyi jelek feldolgozásában.

Ezek a lépések együttesen segítenek megoldani a szakdolgozatomban felvetett problémát, azaz az EEG jelek gépi tanulással történő hatékony feldolgozását és az BCI létrehozását.

3. Irodalomkutatás

3.1. Központi idegrendszer

Az ideg és az endokrin rendszerek összehangolva irányítják a test funkcióit. Az idegrendszer alapvetően speciális sejtekből, neurokból áll, amelyeknek funkciója az érzékszervi információk fogadása és továbbítása az effektor szervekhez. Ezek az effektorok lehetnek izmok vagy mirigyek, feladatuk pedig az, hogy a testben kiváltott ingerekre valamilyen válaszreakcióval feleljenek.

Az idegrendszerünk az érzékszerveken keresztül kapja az információkat, melyek kiváltói lehetnek külső vagy belső ingerforrások. A beérkezett információk ezután az idegrendszerben összehangolódnak úgy, hogy az effektor szerveink kellőképpen harmonikusan funkcionáljanak. [2] Ezt a működést a következő példával szemléltetném: képzeljük el azt a hétköznapi helyzetet, hogy sütés közben véletlen rátenyerülünk a bekapcsolt állapotban lévő tűzforró indukciós sütőlapra. Ekkor a tenyerünkben található idegsejtek észlelik a hőmérséklet rapid, de annál jelentősebb növekedését (az ingert), aminek következtében az idegek továbbítják az üzenetet az idegrendszernek, ami jelez az izmainknak, hogy minél hamarabb húzzák el a kezünket a perzselő felülettől. Ezzel a külső ingerre adott válaszreakcióval megóvjuk a végtagunkat az esetleges sérülésektől vagy maradandó károsodásoktól.

Az agyunk el tudja raktározni a korábbi tapasztalatainkból származó lényegesebb információkat. Ha ezek az információk megfelelő formában állnak elő az idegrendszer rendelkezésére, akkor integrálódnak más idegi jelekkel és közös útvonalon továbbítódnak. Ez nyújt segítséget abban, hogy képesek legyünk effektíven alkalmazkodni a környezetünkhöz.

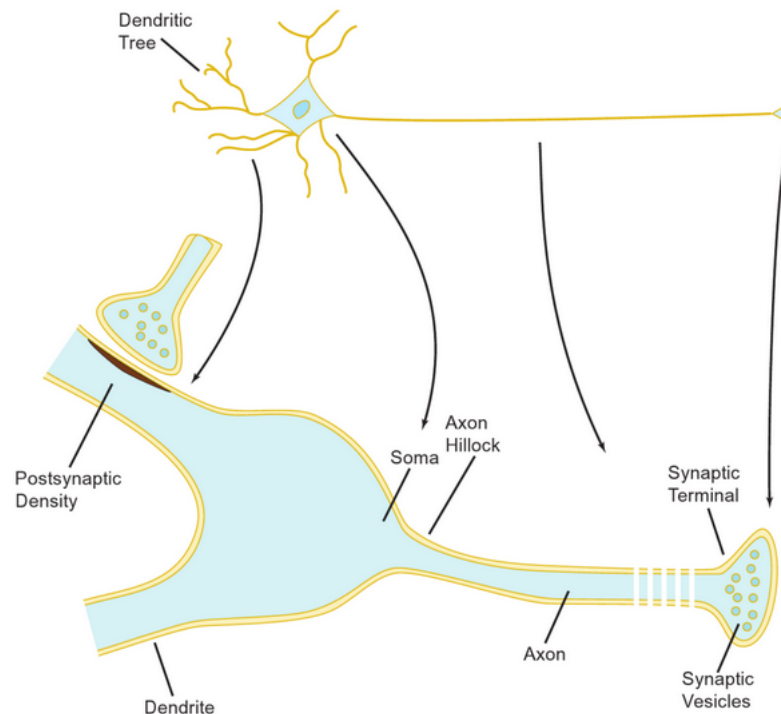
Az idegrendszer két fő részre oszlik: a központi idegrendszerre és a perifériás idegrendszerre. A központi idegrendszer az agyból és a gerincvelőből áll. Ezek a fő központok, ahol az idegi információkat feldolgozzuk és összehangoljuk. A központi idegrendszerben sok „ingerelhető” idegsejt található, amelyeket a neuroglia nevű speciális szövet támogat. A perifériás idegrendszer az agyhoz és a gerincvelőhöz kapcsolódó idegrostokból és ganglionjaikból áll. Ezek az idegrostok információt továbbítanak a központi idegrendszer felé és vissza.

Az idegrendszer mindent összekapcsol, az érzékeléstől az izomműködésen át a válaszreakciókig. Az idegrendszerünk és az endokrin rendszerünk szorosan együttműködik, hogy fenntartsa a testünk egyensúlyát és optimális működését. Amikor érzékeljük az érintést, látványt vagy hangot és reagálunk rá, tudhatjuk, hogy az idegrendszerünk éppen

munkát végez, hogy a testünk kiegyensúlyozottan tudjon működni. [3]

3.1.1. Neuronok

A neuronok olyan sejtek, melyek ingerimpulzusok érzékelésre és azok továbbvezetésére specializálódtak. A neuronok méretben és alakban eltérhetnek, de mindegyikük rendelkezik egy sejtttesttel, amelyből egy vagy több neurit indul ki. Ezek a neuritok olyan még fejlődésben lévő axonok vagy dendritek amelyeket nem lehet egyszerűen megkülönböztetni. A neuritok felelősek az információk fogadásáért és azok továbbításáért a sejttest felé. Ahogy az a 3-1. ábrán is látható az egyetlen hosszúcsövű neurit, amely az impulzusokat elvezeti a sejttestől távolabb, az axon. [3]



3-1. ábra. Az ábrán egy tipikus idegsejt, amin különböző részei vannak megjelölve. [4]

A dendritek az információkat fogadják az idegrendszer többi részétől, majd továbbítják ezeket a sejttest felé. Tehát elképzelhetjük őket, mint a neuronok antennáit, amelyek figyelik a környezetükből érkező jeleket. Az axon pedig a neuron hosszú, cső alakú alkotóeleme, amelyen keresztül az impulzusokat továbbítja más neuronokhoz vagy effektor szervekhez. Ezek a neuron komponensek tulajdonképpen az „idegrendszer vezetékei”, amelyeknek köszönhetően információk gyorsan utazhatnak a neuronok között. [2]

3.2. Agy-számítógép interfész

Pár évtizeddel korábban az agyjelek mérése rendkívül költséges és bonyolult folyamat volt. [5] Azonban az elmúlt években a hardverek és szoftverek fejlődése lehetővé tette, hogy az EEG jelek mérésének és feldolgozásának folyamata számottevően pontosabb, gazdaságosabb és gyorsabb legyen.

Emellett az elmúlt 50 évben jelentős előrelépések történtek a központi idegrendszer működésének megértésében. Az idegtudomány fejlődése révén számos információra tettünk szert, ami jelentősen javította a rövid és hosszú távú EEG jelek elemzését. Ez a tudás lehetővé teszi számunkra, hogy jobban megértsük az agy működését és azt, hogy hogyan kapcsolódik a test többi részéhez.

Ezenfelül az életfenntartó technológiák, mint az otthoni lélegeztetőgépek hozzájárultak ahhoz, hogy az emberek, akik különböző neurális betegségekben vagy rendellenességekben szenvednek, hosszabb és kényelmesebb életet élhessenek. [6] A meghosszabbodott élet-tartamú emberek számára ezek teremtették meg az igényt olyan megoldásokra, amelyekkel ismét képesek kommunikálni, interaktálni a külvilággal sérülésük vagy betegségük után.

Jerry J. Shih szerint egy BCI olyan rendszert jelent, amely képes gyűjteni, analizálni és átalakítani az agyjeleket utasításokká, hogy azok továbbíthatók legyenek bármilyen eszköz vagy rendszer számára. [7]

Az agy fő feladata az, hogy válaszoljon az általunk érzékelt ingerekre. Az ingerek, legyenek azok neuromuszkulárisak vagy hormonálisak, válaszreakciók generálásra készítetik az agyat. A BCI új lehetőségeket nyit meg a központi idegrendszer számára. A BCI lehetővé teszi az agyi aktivitás mérését és mesterséges kimenetek generálását. Az agyi aktivitás elektrofiziológiai, neurokémiai és anyagcsere folyamatokból áll. Ezeket az agyi jelenségeket elektromos vagy mágneses mezők, hemoglobin oxigénszint vagy más paraméterek monitorozásával lehet mérni a fejbőrön, az agy felszínén vagy az agyban elhelyezett szenzorok segítségével.

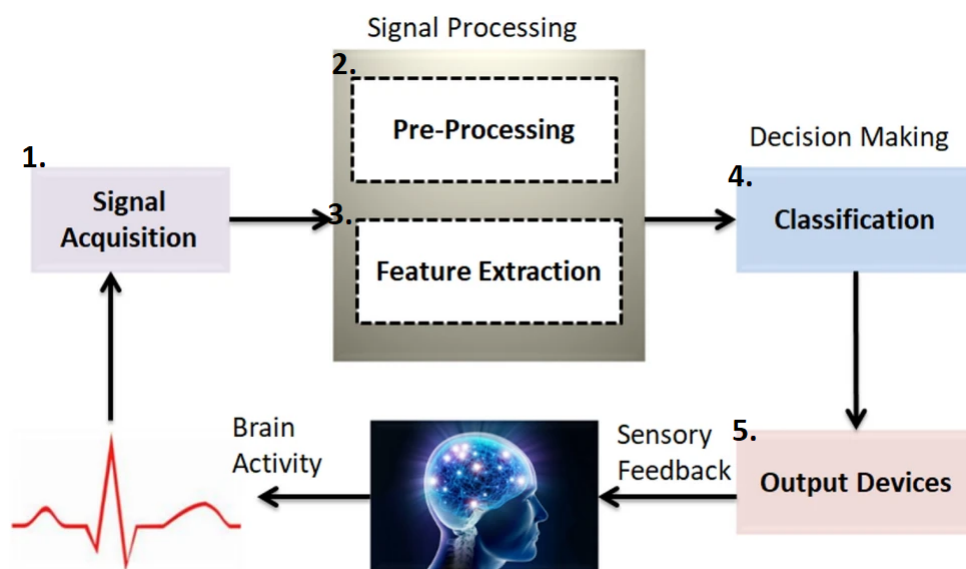
A BCI rögzíti ezeket az agyi jeleket, kiválasztja a jellemzőket, majd ezeket a jellemzőket mesterséges kimenetekké alakítja át. Ezek a mesterséges kimenetek hathatnak a külvilágra vagy akár a testünkre. Ezek a kimenetek képesek helyettesíteni, helyreállítani, fokozni, kiegészíteni vagy javítani a természetes idegrendszeri kimenetet és ezáltal megváltoztatni az idegrendszer és a külső vagy belső környezet közötti kölcsönhatásokat.

Így a BCI egy olyan új lehetőséget kínál, amely az agyi jelek alapján működik és képes

befolyásolni az agy és a külvilág közötti kommunikációt. Ez az innovatív technológia nagy lehetőségeket rejt magában a neurális rendellenességek kezelésében és a funkcionális rehabilitáció területén. [6]

3.2.1. A BCI-k alapelvei

Egy BCI rendszer képes „kommunikálni” az emberi aggyal és végrehajtani a parancsokat, amelyeket a gondolatainkon keresztül küldünk. Ezek a rendszerek öt fő modulból állnak, amelyek mind fontos szerepet játszanak a gondolataink dekódolásában és azok számítógépes parancsokká való alakításában.



3-2. ábra. Egy EEG alapú BCI rendszer működésének lépései az EEG jelek előállítása, ezt követi a gyűjtött adat feldolgozása, majd a jellemzők kigyűjtése. Ez után a jellemzők osztályozásra kerülnek, végül az eredményt valamilyen formában továbbítjuk a BCI-hez kötött kimeneti eszközhöz. [8]

Az első modul a jelgyűjtés (3-2-es ábrán az 1.-es ponttal jelölve). Itt az agyi jeleket rögzítik, erősítik és digitalizálják. Az agyunk különféle jeleket küld, amikor gondolatainkkal vagy mozgásunkkal foglalkozunk. Ezeket az jeleket invazív és non-invazív módszerekkel lehet rögzíteni. Az invazív módszerben tűket vagy szenzorokat helyeznek be az agyba, míg a non-invazív módszerek a fejbőrön vagy annak hajjal fedett részén rögzítik a jeleket.

Az agyi jelek azonban nem mindig tökéletesek, mivel más biológiai folyamatok vagy zajok is befolyásolhatják azokat. Ezért van szükség a második modulra, a jel előfeldolgozásra

(3-2-es ábrán az 2.-es ponttal jelölve). Itt az agyi jeleket tisztítják és javítják a zavaró tényezőktől. Például eltávolítják a zajokat vagy más műtermékeket, amelyek befolyásolhatják az agyi jeleket.

A harmadik modul a jellemzők kinyerése (3-2-es ábrán az 3.-as ponttal jelölve). Itt egy-egy vizsgálat során, olyan jellegzetességeket választanak ki az agyi jelekből, amelyek általánosak és relevánsak az agy működésére nézve. Két fő típusú jellemzőt használnak: frekvencia-teljesítmény jellemzőket és időbeli jellemzőket. Az előbbiek azt mutatják meg, hogy az agyi jelek milyen frekvenciasávban vagy ritmusban aktívak, míg az időbeli jellemzők az agyi jelek időbeli változásait veszik figyelembe a vizsgálat során.

A következő a negyedik modul, a jellemzők osztályozása (3-2-es ábrán az 4.-es ponttal jelölve). Itt különböző algoritmusokat használnak az agyi jellemzők felismerésére és osztályozására. Ilyen célokra a lineáris algoritmusokat vagy mély neurális hálókat szokás alkalmazni. Az osztályozás során az agyi jeleket számítógépes parancsokká alakítjuk át.

Végül az ötödik modul a kimeneti eszközök (3-2-es ábrán az 5.-ös ponttal jelölve), amelyeken keresztül a parancsok végrehajtódnak. Ezek az eszközök lehetővé teszik számunkra, hogy a valós világban cselekedjünk, anélkül, hogy közvetlenül a perifériás idegeinket és izmainkat használnánk. Például funkcionális elektromos stimulációt (FES) [9] vagy protéziseket alkalmazhatunk a mozgás vezérlésére.

Az FES az elektromos impulzusokkal stimulálja az izmokat, amelyek segítségével az agy kap visszajelzést a mozdulatainkról. Egy másik példa egy robotikus protézis, amely képes helyettesíteni egy elveszett végtagot, így egy kart is. Ezek az eszközök nemcsak a cselekvést teszik lehetővé, hanem visszajelzést is nyújtanak az agynak, hogy finomíthassa a mozgásokat a sebesség és pontosság javítása érdekében. A BCI rendszerek forradalmi módon lehetővé teszik a gondolataink használatát a külvilág irányítására. Ez az új technológia nagy lehetőségeket rejt magában az idegi rehabilitációban és más területeken. [9]

3.2.2. Invazív és non-invazív módszerek

Az agyi jelek rögzítésének két fő módja van: invazív és non-invazív. Ahogy arról az előző fejezetben is kitértem, az invazív módszer során kis szenzorokat vagy tűket helyeznek az agyba, hogy közvetlenül felvegyék az agyi jeleket. A non-invazív módszerek pedig kívülről rögzítik az agyi jeleket, anélkül, hogy bármit be kellene helyezni az agyba. Az elektroencefalográfia gépek segítségével rögzíthetjük az agyi jeleket és ezek a BCI rendszerek képesek nagyon részletesen feldolgozni a jeleket a számítógépek segítségével.

Az EEG gépek olyan eszközök, amelyek speciális elektródákkal rendelkeznek, amelyek pontosan mérni tudják az agyi jeleket. Ezek a gépek rendelkeznek számítástechnikai rendszerekkel és elegendő memóriával, hogy hosszú távú felvételeket készítsenek, amelyek akár több órák is lehetnek. Így részletesen és folyamatosan megfigyelhetjük az agy működését és aktivitását hosszú időn keresztül.

Amikor az agyi jelekről beszélünk, fontos egy kitérőt tennünk, hogy jobban megismerhessük az invazív agyi rögzítést. Ez a módszer arra szolgál, hogy közelebb kerüljünk a neuronokhoz, az agy sejtjeihez és így pontosabb információt szerezzünk a működéséről és esetleges hibás területeiről. Az ilyen módszerrel például mély agyi stimulációt, sebészeti beavatkozást lehet végezni vagy rehabilitációs implantátumokat elhelyezni. Itt a kutatók finom tűszerű elektródákat használnak az agy fölött, hogy közvetlenül rögzítsék az agyi elektromos aktivitást. Ezáltal elkerülhetők a koponya okozta torzítások és nemlineáris hatások, így lehető legpontosabb eredményeket kaphatjuk meg.

Ez a közvetlen agyi rögzítés, amit elektrokortikográfiának (ECoG) vagy intrakraniális elektroencefalográfiának (iEEG) nevezünk, egy olyan módszer, amely az agykéregnek közvetlen felületéről rögzíti az elektromos aktivitást. Ezt úgy kell elképzelni, mint egy finom elektródaszerű rácsot, amelyet közvetlenül az agy felületére helyeznek és ezen keresztül rögzítik az agyi jeleket.

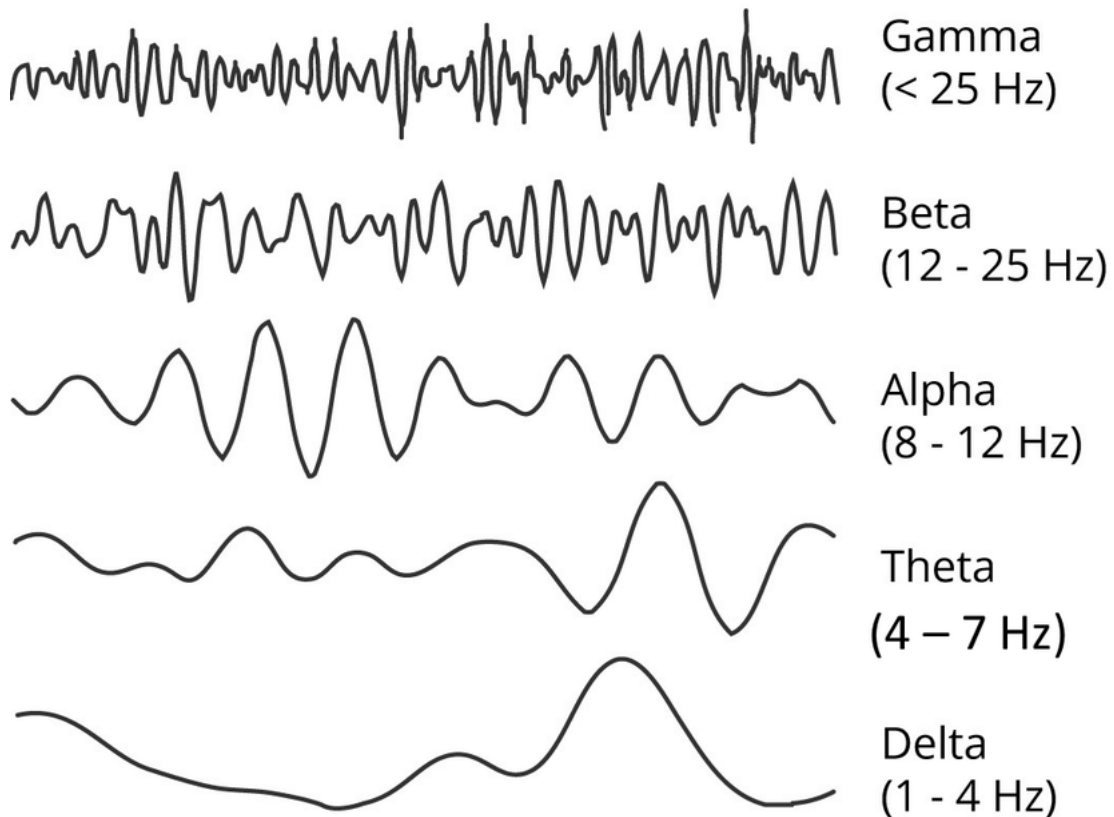
Az invazív módszerek a pontosabb méréseket eredményeznek ami annak köszönhető, hogy „közelebbről” tudják az agyhullámokat mérni. Azonban az alkalmazásuk velejárója, hogy rendkívül kockázatosak és körülményesek. Ezért én a szakdolgozatomat a non-invazív oldalról közelítem meg. Választásom az EEG-re esett, mert ez a fajta non-invazív mérés nem igényel sok előkészületet, biztonságosabb és költség-, illetve időhatékonyabb.

3.3. Elektroencefalográfia

Az EEG mérés során kis elektródákat helyeznek a fejbőrre, vagy annak hajjal fedett részére. Ezek az elektródák érzékelik az agyból származó elektromos jeleket, amelyeket az idegsejtek generálnak. Az agyi aktivitás ezáltal elektromos potenciálokká alakul, amelyeket az EEG képes mérni.

Az EEG értékes információkat nyújt az agyműködésről és széles körben használják orvosi és kutatási környezetben. Segít tanulmányozni és megérteni különböző neurológiai állapotokat és rendellenességeket. Az EEG nagy előnye, hogy nagyon érzékeny az agyi aktivitás változásaira és az időben is részletes információkat nyújt. Míg más képalkotó módszerek, mint például a funkcionális mágneses rezonancia képalkotás (fMRI) vagy a

közel-infravörös spektroszkópia (NIRS), másodperceken vagy perceken belül működnek, az EEG ezeket a változásokat akár milliszekundumos időskálán is észleli.



3-3. ábra. A képen különböző típusú agyhullámok láthatóak, melyek amplitúdója és frekvenciája eltérő: gamma, béta, alfa, teta és delta hullámok. Ezek az agyhullámok különböző agyi állapotokat és funkciókat tükröznek. [10]

Az elmúlt 100 évben a neurobiológusok és a klinikai neurológusok széles körben használták az EEG-t. Segítségével megismerhették a kognitív folyamatokat és az agyi rendellenességeket. Az EEG jelentőségét azonban nemcsak a kutatók és orvosok ismerik fel. Az utóbbi időben egyre gyakoribb, hogy a BCI-ket az EEG felhasználásával valósítanak meg. Ez azt jelenti, hogy az a EEG feldolgozásból szerzett információkat használhatjuk arra, hogy közvetlenül irányítsunk fizikai eszközöket az agyunk segítségével.

Agyunk működésének elemzésében az agyhullámok kulcsfontosságú szerepet játszanak. Ezek az elektromos impulzusok az agysejtek aktivitásából erednek, és eltérő frekvenciájúak és jellegzetességeik alapján különböztetjük meg őket. Ezeket a hullámokat a 3-3-es ábrán láthatjuk.

A leglassabb frekvenciájú hullámok a delta hullámok. Ezek jellemzően 0,5-4 Hz között mozognak. Általában mély alvás vagy kóma állapotában jelennek meg.

A theta hullámok alacsony frekvenciájú hullámok, amelyek általában 4-8 Hz között mozognak. Jellemzően álmoság, mély relaxáció vagy meditáció közben jelentkeznek.

Alfa hullámoknak nevezzük a 8-12 Hz közötti intervallumot. Ezek a hullámok nyugodt állapotban vagy zárt szemmel végzett relaxáció közben fordulnak elő.

A 13-30 Hz között hullámokat beta hullámoknak hívjuk. Ezeket az éber tudatállapothoz, a figyelemre, a gondolkodásra és a mentális tevékenységekre.

A gamma hullámokat 30 Hz-től általában 100 Hz-ig szoktuk mérni. Gyakran kapcsolódnak a magas szintű agyi működéshez, például az információfeldolgozáshoz, a memóriához és a tudatossághoz.

Záró gondolatként érdemes említést tennünk az artefaktumokról. Az EEG jelekben található műtermékek vagy másnéven artefaktumok olyan nem kívánt jelzések, amelyek torzíthatják az EEG adatok interpretálását. Ezek a jelek belső és külső forrásokból egyaránt származhatnak, ideértve a szemmozgásokat, az izomműködést, a szívverést és a környezeti zajokat. Például a szemmozgásokból eredő artefaktek magukban foglalják a pislogások és egyéb szemmozgások által keltett elektromos jeleket, míg az izomartefaktek főként a fej, az arc és a nyak izmainak aktivitásából erednek. Ezeket az artefakteket számos módszerrel lehet csökkenteni vagy eliminálni, mint például az elektródák gondos elhelyezésével, a vizsgálati személynek adott részletes utasításokkal, valamint kifinomult jelzésfeldolgozási technikákkal.

Fontos megjegyezni, hogy ezek az agyhullámok nem egymástól elszigetelt jelenségek, hanem átfedésben és kölcsönhatásban vannak egymással. Ezek vizsgálata segíthet megérteni az agyműködés különböző aspektusait.

3.3.1. Motoros paradigma

Az olyan elképzelést, amit fejben formálunk meg, de fizikailag nem valósítjuk meg, mozgás-képalkotásnak nevezzük. Amikor elképzeljük a mozgást, a szenzomotoros hullámok, azaz az agy elektromos aktivitásának bizonyos tartományai, mint a mu (8-12 Hz) és béta (13-30 Hz) agyhullámok megváltoznak. [11]

A mozgás-képzet alapú BCI rendszerek képesek észlelni ezeket a változásokat és azo-

kat átfordítani irányítási utasításokká külső eszközök vagy alkalmazások számára. Ez azt jelenti, hogy az elképzelt mozgások képesek vezérelni fizikai eszközöket, például protéziseket vagy számítógépes alkalmazásokat. [12]

A mentális képalkotás a vizuális és kinetikus képzeletet foglalja magában. Fontos megjegyezni, hogy a sikeres mentális képalkotás nagy részben a végre hajtandó feladattól függ. Az egyik leggyakoribb utasítás fajta, amit a résztvevők általánosságban kaphatnak az az, hogy képzeljék bele magukat a feladat által meghatározott környezetbe és hajtsák végre a kapott utasításokat a képzeletükben.

A mentális képalkotásnak három komponense van: élénkség, irányíthatóság és pontos sablonreferencia-kép. Az élénkség azt jelenti, hogy a kép élesen és élethű jelenik meg elménk képzelet belüli vásznán. Az irányíthatóság azt, hogy az alany képes legyen pontosan navigálni a képet az elképzelése alapján. A pontos sablonreferencia-kép pedig segít abban, hogy megőrizze a kép részleteit és struktúráját. [13]

3.3.2. Állandó állapotú vizuálisan kiváltott potenciálok

Az állandó állapotú vizuálisan kiváltott potenciálok (SSVEP-k) [14] olyan feszültség ingadozások, amelyeket gyors ismétlődő stimuláció váltanak ki, mint amikor egy stroboszkóp fénye gyorsan villog, vagy egy LED vagy egy sakktábla mintázatot és annak színek szempontjából invertált változatát akarjuk megjeleníteni a monitoron. Ezek az ismétlődő ingerek SSVEP-eket generálnak. Az SSVEP-eket különböző analitikus módszerekkel vizsgálhatjuk. Ezen belül a leggyakoribb megközelítések az átlagolás vagy frekvenciaanalízis. A frekvenciaanalízis általában csúcsokat mutat a stimuláció frekvenciájánál.

A figyelemszint paradigma olyan kutatási irány, amely a figyelem és koncentráció mélyebb megértését célozza meg az agyi aktivitás változásain keresztül. Először is, az agyban található alfa hullámokat vizsgáljuk, amelyek egy bizonyos frekvencián (8-12 Hz) rezegnek. [15] Érdekes módon amikor az alfa aktivitás növekszik, az általában egy relaxált állapothoz társul. Tehát, ha elmélyedtünk egy könyv olvasásában vagy egy pihentető széken ülünk, valószínűleg az agyunk alfa hullámai megnövekednek.

Az alfa hullámok aktivitásának csökkenése azt jelenti, hogy a vizsgált alany fokozott figyelmet vagy koncentrációt igénylő feladatokat lát el. Azért fontos ez a felismerés, mert a figyelemszint paradigma segítségével nyomon követhetjük az alanyok mentális állapotát és ezáltal a BCI rendszerünket is az igényeikhez igazíthatjuk. Például a BCI visszajelzést adhat a felhasználó számára, hogy segítse a koncentráció fenntartását, vagy akár automatikusan módosíthatja a feladat nehézségét a figyelemszintjének megfelelően. [15]

Az EEG által kiváltott potenciálok azok a pillanatok az agyban, amikor figyelmünket egy rögzített szituációban az érzékszervi ingerekre vagy eseményekre irányítjuk. Ezek időben zárolt feszültségváltozásokat generálnak, melyek pozitívak és negatívak is lehetnek. Vizuális ingerek esetén, amikor hirtelen megjelenik egy fényvillanás, egy kép vagy egy színváltás, ezeket a feszültségváltozásokat vizuálisan kiváltott potenciáloknak nevezzük. [6]

Egy szabványos SSVEP-alapú BCI rendszerben a felhasználó több ismétlődő ingert (például több LED-et) lát egy kijelzőn, amelyek különböző helyeken vannak a látóterében. Minden inger más frekvencián jelenik meg, ami megkülönbözteti azokat egymástól. Minden inger egy adott kimenetet képvisel a BCI rendszerben, ilyen lehet egy adott betű begépelése vagy egy kerekesszék egy bizonyos irányba való mozgatása.

A felhasználó egyszerűen azzal választ, hogy vizuálisan rögzíti más szóval erősen koncentrálva nézi azt az ingert, amelyik a kívánt kimenetet képviseli. Az agyban az EEG jelek egy általunk kiválasztott tartományát vizsgáljuk és általában egy csúcsot találunk, amely megegyezik a felhasználó által rögzített inger frekvenciájával. Ezután a BCI előállítja az adott kimenetet, amelyet az a specifikus inger vált ki.

Ezeket a szabványos SSVEP BCI rendszereket f-VEP (frekvencia-modulált vizuálisan kiváltott potenciál) BCI paradigmátípusnak nevezik. Azonban léteznek más paradigmák is. Itt az átlagos VEP-t számítjuk ki minden egyes ingerre és az adott inger által kiváltott kimenetet generáljuk. Van még a c-VEP (vagy m-sorozat) paradigmája, ahol az ingerek pseudo-random mintázatban jelennek meg, ami szinte ortogonális a többi inger mintázatához képest. Ebben az esetben az EEG és az egyes ingerek közötti korrelációt számítjuk ki. Általában a felhasználó tekintetét követő ingert érzékeljük a legmagasabb korrelációval. [6]

A jövőben a SSVEP-alapú BCI rendszerek tovább fejlődhetnek és vizsgálhatják a c-VEP és más paradigmákat is. Ezek az alternatívák technikailag nem SSVEP paradigmák, mert az ingerek nem ritmikusak, de kényelmi okokból általában SSVEP BCI-ként csoportosítják őket. [6]

3.3.3. Eseményhez kötött potenciál

Az agyunk működését tanulmányozva felfedezhettük, hogy rendkívül érzékeny és különleges reakciókra képes. Az EEG segítségével nyomon követhetjük az agyunkban zajló eseményeket és megérthetjük, hogyan válaszolunk ezekre. Az agyi reakciókat esemény

vezérelt potenciáloknak (ERP) nevezzük. Ezek az agy elektromos aktivitásának változásai, amelyeket az események váltanak ki. Ezeket a potenciálokat általában az EEG-felvételeken rögzítjük. [16]

Az ERP-k időben „rögzülnek” az eseményekhez és követik a folyamatot, ami azt jelenti, hogy nyomon követhetjük az agy reakcióit az időbeli lefolyásukon keresztül. Azáltal, hogy többször is megismételjük az eseményt és átlagoljuk a felvett válaszreakciókat, kivonhatjuk azokat a jelenségeket, amelyek az eseményekhez kapcsolódnak. Így kiemelhetjük az agyhullámokban található komponenseket.

Ezek a komponensek a pozitív és negatív feszültség szélsőértékek sorozatai, amiket különböző stimulusok váltanak ki. A komponensek különböznek egymástól az amplitúdójukban, a topográfiájukban és az agy információfeldolgozási folyamataival való kapcsolatban. Az agyunk reakciói sokféle formában jelentkezhetnek attól függően, milyen ingerek érik.

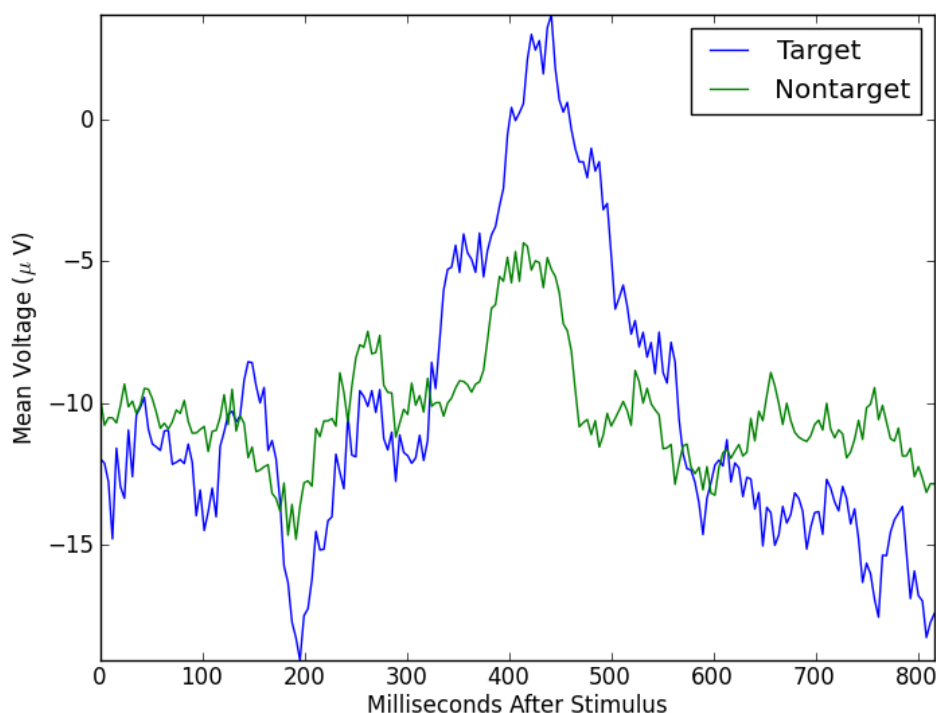
Vannak korai komponensek amelyek az események után jelentkeznek az első 150 milliszekundumban és általában a primer érzékszervi rendszereink aktivitását tükrözik. Az ezekhez kapcsolódó hullámformák és a fejbőrön megjelenő aktivitás eloszlása változik a különböző ingerektől függően. Ezeket exogén komponenseknek nevezik.

A hosszabb idő után jelentkező komponensek inkább a kognitív folyamatokhoz kapcsolódnak. Ezek inkább függenek az aktuális feladatok jelentőségétől, mint az ingerek jellegétől. Amikor egy kognitív feladatot végzünk, az agyunk ezekkel az endogén komponensekkel reagál.

3.3.4. A P300-as jel

A P300 hullám (vizuálisan 3-4-es ábrán) olyan az agyban létrejövő pozitív változás, ami figyelem irányításában, a kontextus frissítésében, a válaszmodulációban és a válaszfelbontásban játszik szerepet. A név onnan ered, hogy a stimulus kiváltása után 300 ms-al jelenik meg egy pozitív irányba történő csúcsosodása az EEG hullámnak. Két fő alkotóelemből áll: a P3a és a P3b. A P3a az új vagy váratlan ingerekre adott automatikus figyelemirányítást tükrözi, függetlenül a feladat relevanciájától. A P3a generálásában a prefrontális, frontális és elülső időbeli agyterületek játszanak szerepet. Ezzel szemben a P3b nagyobb centro-falcsonti eloszlást mutat és a hátsó temporális, falcsonti és hátsó cinguláris kéreg mechanizmusaira támaszkodik. [18]

Ezeket a hullámokat és az agy reakcióit az ERP-k és az EEG segítségével tanulmányozzák. Az ERP-k az agyunk a különböző új vagy váratlan ingerekre adott válaszát mérik. Ez az



3-4. ábra. A grafikon az agyi válaszokat ábrázolja, ahol a kék görbe azonosítja a „célzott” ingerekre adott elektrofiziológiai reakciókat (P300 hullámokat), míg a zöld görbe a „nem célzott” ingerekre adott válaszokat mutatja. [17]

ERP nem csak vizuálisan idézhető elő, hanem audio ingerek formájában is. Ha egy hang meglepő vagy ismeretlen, akkor nagyobb P3a választ válthat ki, mint egy ismerős hang. Hasonlóképpen, a P3b ritka eseményekre reagál, de már a feladattal vagy döntéssel kapcsolatos komponenst is tartalmaz. [19]

Az orientációs válasz, amelyhez a P300 tartozik, az agyunk automatikus figyelemeltolódása új vagy váratlan ingerre. Ez egy gyors válaszmechanizmus, ami fontos szerepet játszhat a túlélésünkben. Az orientációs válasz azt mutatja, hogy az ismételt bemutatás után azok az ingerek amelyek először választ váltottak ki, már nem teszik ezt meg. Ez a válasz eredménye és azt sugallja, hogy az agyunk valamilyen formában eltárolta ezeket az eseményeket és módosítja a választ az ismételt ingerlésre. [20]

A P300-ban található rendellenességek számos pszichiátriai és neurológiai állapotban megfigyelhetők, bár azok hatása a P3a-ra és a P3b-re eltérő lehet. Az audio- és vizuális P300-at különböző módon érintik ezek az állapotok és más módon reagálnak a kezelésre is. Ez részben annak köszönhető, hogy különböző szerkezetekben és neurotranszmitter

rendszerekben nyilvánulnak meg. [18]

Ezek alapján jól látható, hogy a P300-nak jelentős diagnosztikai és prognosztikai potenciálja van, különösen akkor, ha más klinikai tünetekkel kombinálják. Például az ember-számítógép interakcióban, az izomfáradtság, a vizuális fáradtság és a mentális fáradtság vizsgálatában használják.

3.4. Gépi tanulás

A gépi tanulás a mesterséges intelligencia és a számítástudomány egy ága, amely az adatok és algoritmusok felhasználására összpontosít annak érdekében, hogy utánozza az emberi tanulás módját és fokozatosan javítsa pontosságát.

Arthur Samuel, az IBM kutatója a gépi tanulás kifejezés megalkotója volt és dámajáték körüli kutatásaival vált ismertté. 1962-ben a magát dāmamesternek nevező Robert Nealey, egy IBM 7094 számítógéppel mérkőzött meg dámajátékban és meglepő módon veszített. Bár ez ma már jelentéktelennek tűnhet, az akkori időben ez egy fontos mérföldkőnek számított a mesterséges intelligencia területén. [21]

Az elmúlt évtizedekben a tárolási és feldolgozási kapacitások fejlődése lehetővé tette néhány innovatív termék létrehozását a gépi tanulás alapján. A Netflix például a gépi tanulást használja ajánló rendszerében, amely képes személyre szabott ajánlásokat adni a felhasználóknak a nézett tartalmak alapján. **netflix** Az önvezető autók is a gépi tanuláson alapulnak és képesek tanulni és javítani vezetési képességeiket az adatok és algoritmusok felhasználásával. [21]

A gépi tanulásnak fontos szerepe van az adattudomány területén is. Az algoritmusok statisztikai módszerekkel képesek osztályozni vagy előrejelzéseket tenni az adatok alapján és kulcsfontosságú betekintést nyújtanak az adatbányászati projekteken. Ezek az adatok az alkalmazások és vállalkozások számára nyújtanak segítséget a döntéshozatal során.

A gépi tanulási során a leggyakrabban használt keretrendszerek TensorFlow és a PyTorch. [22] Ezek a keretrendszerek felgyorsítják a megoldások fejlesztését és lehetővé teszik a modellek építését az adatok alapján. Például a felügyelt gépi tanulás használatával modelleinket tanítjuk történelmi példák vagy példányok alapján, ahol az algoritmusok automatikusan megtanulják a leíró jellemzők és a cél jellemző közötti kapcsolatot.

Azonban az adatokkal való összhangban lévő modellek keresése nem elegendő a hasznos előrejelzési modellek tanulásához. Nagy adathalmazok esetén gyakran található zaj az

adatokban, emiatt a modellek helytelen előrejelzéseket hozhatnak létre. [18]

A gépi tanulás magában foglalja matematikai modellek és algoritmusok használatát a megoldás teljesítményének fokozása érdekében. Két fő típusa van: a felügyelt és a felügyelet nélküli tanulás. A felügyelt tanulás bemeneti és kívánt kimeneti adatokat használ az osztályozó vagy regressziós modellek tanításához. A felügyelet nélküli tanulás pedig csak bemeneti adatokat használ és klasszifikációra vagy az adatpontok reprezentálására használt dimenziók számának csökkentésére alkalmazható.

Összességében a gépi tanulás az adatok és algoritmusok használatával utánozza az emberi tanulás folyamatát. A példák és adatok alapján épített modellek segítségével előrejelzéseket és döntéseket hozhatunk, amelyek változásokat hozhatnak számos területen, mint például a filmajánlás és az önvezető autók esetében. Azonban a zajos adatok és az alacsony minta számok olyan kihívásokat jelentenek, amelyekre figyelmet kell fordítani a megbízható és pontos eredmények elérése érdekében.

3.4.1. A gépi tanulás az EEG területén

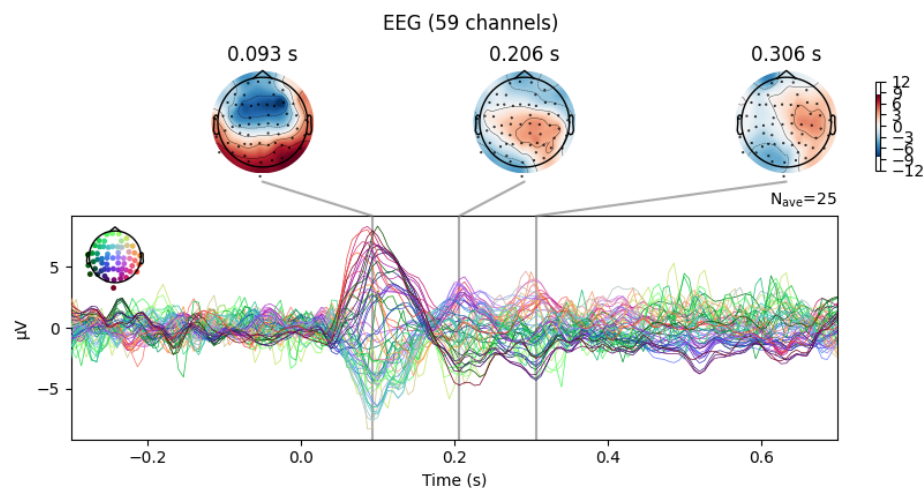
Ahogy az előző szekcióban kifejtettem, a gépi tanulás olyan folyamat, amely az adatokból történő tanulásra és adatok osztályozására épül. Az EEG olyan mérésekre használható, amelyek az agyi elektromos aktivitását rögzítik. A gépi tanulás lehetőséget nyújt arra, hogy az EEG jelekből kiemeljük az agyi jellemzőket. [23] Ezenkívül hatékonyságának növelése érdekében gyakran kombinálják különböző jelfeldolgozási technikákkal, amelyek elősegítik a legfontosabb adatjellemzők kiemelését a tanulási és osztályozási folyamatok előtt.

A gépi tanulás folyamatában szerepel a jelek rögzítése, az előfeldolgozás és a jellemzők kinyerése. A jellemzők kiválasztása olyan új attribútumok létrehozását foglalja magában, amelyek összegzik a korábbi adatok jellemzőit. A tanulási fázisban a gépi tanulási algoritmusok finomítják az osztályozási módszereket a tanulási adatkészletek segítségével. Végül a tesztelési szakaszban valós adatkészleteket használnak fel az algoritmusok pontosságának értékelésére.

Az EEG jelek és a gépi tanulás együttes alkalmazása lehetővé teszi, hogy ezek együttes kombinációját biomarkereként használjuk bizonyos nehezen diagnosztizálható orvosi állapotok esetén. A gépi tanulás segítségével kifejleszthetők előrejelző modellek, amelyek nem csak osztályozásra, hanem regresszióra is alkalmasak, így támogatva a diagnózis és kezelés folyamatát. [18]

Az EEG adatok elemzéséhez és feldolgozásához számos eszköz áll rendelkezésre, amelyek különböző szempontból támogatják a kutatók munkáját. Az egyik legnépszerűbb ilyen eszköz az EEGLAB, amely egy nyílt forráskódú, MATLAB-környezetben működő platform. Széles körű alkalmazhatósága és modularitása lehetővé teszi, hogy a kutatók testre szabhassák és bővíthessék a rendszer funkcionalitását. Az EEGLAB elősegíti az adatok előzetes feldolgozását, az artefaktumok kiszűrését, az adatok csoportosítását, a frekvencia analízist és különféle statisztikai tesztek elvégzését.

A kutatási folyamat további finomításához az MNE-Python [24] (3-5-as ábra) nyújt kiváló támogatást, amely egy Python nyelven írt eszköz, speciálisan a magnetoencefalográfiai (MEG) és az EEG adatok elemzésére tervezve. Ez az eszköz az adatok előkészítésétől kezdve a zajcsökkentésen át az idő-frekvencia analízisig, valamint a forrás lokalizáció és az agyi hálózatok közötti kapcsolatok vizsgálatáig széles körű funkcionalitást biztosít. [24]



3-5. ábra. Az MNE-vel történő analízis egy lehetséges vizualizációja. [24]

A folyamatban a Brainstorm [25] további vizuális és elemző eszközöket kínál. Ez a szintén nyílt forráskódú, MATLAB-alapú alkalmazás különösen alkalmas a MEG és EEG adatok vizualizációjára és részletes elemzésére. Egyszerűen kezelhető felületén keresztül támogatást nyújt az agyi aktivitás forrásainak pontos becslésére és az agyi hálózatok dinamikájának megértésére.

A FieldTrip [26], egy további MATLAB eszközkészlet, az EEG, MEG és egyéb biológiai jelek elemzésére szolgál, különösen erős a komplex adatstruktúrák kezelésében és a hálódó statisztikai módszerek, mint például a klaszter alapú tesztek alkalmazásában. Ez az eszköz különösen hasznos lehet olyan kutatási környezetekben, ahol az adatok strukturális összetettsége és a statisztikai analízis mélysége kulcsfontosságú.

Az OpenViBE [27], egy platformfüggetlen, nyílt forráskódú program, amely az EEG adatok rögzítésére és feldolgozására szolgál, különösen előnyös a valós idejű BCI fejlesztése során. Alkalmazása ideális biofeedback, neurofeedback és egyéb BCI alkalmazások esetén, ahol a gyors és hatékony adatfeldolgozás elengedhetetlen.

Végezetül a PyEEG [28], egy egyszerű, de hatékony Python modul, amely számos EEG elemzési funkciót kínál, mint például a spektrális analízis, az entrópia számítások és az idő-frekvencia elemzések. Ez a modul lehetővé teszi az agyi aktivitás alapos vizsgálatát, ami esszenciális az agykutatásban.

Ezek az eszközök együttesen biztosítják az EEG adatok széleskörű elemzésének és értékelésének lehetőségét, támogatva a tudományos közösséget a neurológiai kutatások előmozdításában.

3.4.2. A gépi tanulás a P300 területén

A P300-alapú BCI-k olyan technológiák, amelyek lehetővé teszik a felhasználók számára, hogy a környezetüket pusztán az agyi aktivitásuk, konkrétan a P300 jel alapján irányítsák. Ezek a rendszerek nem igényelnek sebészeti beavatkozást, könnyen hordozhatóak, és megbízhatóan működnek, így különösen hasznosak lehetnek azok számára, akik súlyos fizikai korlátozottságokkal élnek, segítve őket a mindennapi életükben. A P300 jel pontos felismerése és az azt követő jellemzők becslése létfontosságú a rendszer hatékonysága szempontjából. Számos kutatócsoport dolgozik azon, hogy tovább finomítsák ezeknek a BCI gépeknek a funkcióit, fokozzák hasznosságukat, és vizsgálják klinikai alkalmazásukat, valamint a technológia további fejlesztési lehetőségeit. [18]

A neurológiai kutatások és klinikai diagnosztika során nélkülözhetetlen az EEG és az ERP adatok részletes elemzése. E célból több, speciálisan erre a célra tervezett megoldás áll rendelkezésre.

A már korábban bemutatott EEGLAB-hoz kifejlesztett ERPLAB bővítmény az ERP adatok részletes elemzésére specializálódott, lehetővé téve az adatok csoportosítását, átlagolását és komponens analízisét. Ez a funkcionalitás fokozza a kutatási eredmények megbízhatóságát, különösen a P300 jellegű eseményvezérelt potenciálok esetében, amelyek kritikusak a kognitív folyamatok megértésében.

A BrainVision Analyzer [29], egy kereskedelmi szoftver, kiterjedt elemzési lehetőségeket kínál az EEG és ERP adatokhoz. Ez magában foglalja az adatok előkészítését, a zajszűrést

és statisztikai analízist, amelyek hozzájárulnak a diagnosztikai eredmények pontosságához és a kutatási adatok alapos kiértékeléséhez.

Végül az előző alfejezetben megemlített MNE-Python, amely szintén támogatja az ERP adatok, így különösen a P300 jelek elemzését. Az adatok előkészítésétől a forrásbecslésen át a statisztikai elemzésig számos funkciót kínál.

3.4.3. EEG adatokat osztályozó algoritmusok

Az EEG területén alkalmazott osztályozó algoritmusok jelentős előrelépést kínálnak a BCI-k fejlesztésében. Ezek a módszerek nem csupán a technológiai hatékonyságot növelik, hanem a felhasználói élményt is javítják, lehetővé téve az egyéni igényekhez való finomhangolást és a felhasználói kényelmet. Az adaptív, tensor alapú és Riemann geometria alapú osztályozók kifejlesztése és optimalizálása továbbra is kulcsfontosságú a pontosabb és megbízhatóbb EEG alapú rendszerek megvalósításához, amelyek képesek kezelni az emberi agy összetett jelzéseit és reakcióit.

Az adaptív osztályozók kiemelkednek azzal a képességükkel, hogy dinamikusan frissítik paramétereiket az új EEG adatok alapján, így képesek alkalmazkodni az adatok változásaihoz. Ez különösen előnyös mikor kevés tanító adat áll rendelkezésre, mivel csökkentheti a további, nagy mennyiségű adatok iránti igényt. A mátrix- és tenzorosztályozók, beleértve a Riemann-geometriai módszereket és a mélytanulási technikákat, egy integrált megközelítést alkalmaznak, amely egyetlen lépésben kezeli az EEG adatok jellemzőinek kinyerését, kiválasztását és osztályozását. Az átadási tanulás módszerei tovább bővítik ezt a képességet azáltal, hogy lehetővé teszik a jellemzők és osztályozók átvitelét különböző domének között, ami hatékonyan csökkenti az EEG adatok non-stacionaritásából adódó problémákat. [30]

A fentiek mellett több konkrét modell is szerepel, mint például a Lineáris diszkriminanciaanalízis (LDA) [31] és annak adaptív változatai, a Tenzoros támogatógép (TSM), a Tenzor Fisher diszkrimináns analízis (TFDA), a magasabb rendű diszkrimináns analízis (HODA), valamint a Riemann geometria alapú módszerek. Ezen módszerek között kiemelkedik a Riemann közép távolság módszere (RMDM), amely különösen hasznosnak bizonyult az EEG adatok összetett jellegének kezelésében. Ezek a technikák együttműködve jelentősen javítják az EEG alapú BCI-k teljesítményét, fokozva az adatok pontosabb osztályozását.

Külön kiemelném az LDA-t aminek a fő célja, hogy azonosítsa azokat a jellemzőket az adathalmazban, amelyek hatékonyan különböztetik meg az adott osztályokat vagy cso-

portokat. Az algoritmus először az adott osztályok átlagait számítja ki, majd egy diszkriminációs funkciót hoz létre, ami minimalizálja az osztályokon belüli varianciát, miközben maximalizálja az osztályok közöttit. [31]

Az LDA egy kiemelkedően teljesítő változata a Shrinkage LDA (sLDA) modell. Az sLDA a kovarianciamátrixok zsugorításával működik, amelynek köszönhetően jobb eredményeket ért el, mint a hagyományos LDA, és az is megfigyelhető, hogy kevesebb tanító adattal is képes hasonlóan magas szintű teljesítményre. Ez az osztályozó különösen előnyös lehet a BCI rendszerekben, ahol gyakran korlátozott mennyiségű adat áll rendelkezésre. [31]

3.4.4. EEG adatok feldolgozása neurális hálókkal

Az EEG adatok elemzése és osztályozása a mély tanulási architektúrák egyik fontos alkalmazási területe, amely jelentős előrelépést jelent a biomedikai kutatások és alkalmazások számára. Ebben a fejezetben az EEG adatok feldolgozására alkalmas neurális hálózatokat vizsgálunk.

Az EEGNet [9] egy olyan mély tanulási architektúra, amelyet kifejezetten az EEG adatok elemzésére fejlesztettek ki. Az architektúra célja egy könnyen alkalmazható, viszonylag kevés paramétert igénylő rendszer kialakítása volt, amely korlátozott adatmennyiség és számítási kapacitás mellett is képes hatékonyan működni. Az EEGNet kulcsfontosságú rétegei, beleértve a konvolúciós és mélységi konvolúciós rétegeket, különösen alkalmasak a térbeli és időbeli jellemzők kiemelésére, amelyek segítségével mélyebb betekintést nyerhetünk az agyi állapotokba. [9]

Az EEGNet adaptivitása lehetővé teszi, hogy széles körben alkalmazható legyen különböző kognitív kutatási és orvosi diagnosztikai feladatokban, valamint a BCI-k fejlesztésében. A hálózat képes kezelni az EEG adatokkal gyakran járó zajokat és eltéréseket, így csökkenti a szükséges előfeldolgozási lépések számát, miközben javítja az adatok megbízhatóságát és pontosságát. [9]

Érdemes megemlíteni még a DeepConvNet-et [32] és a ShallowConvNet-et [33], amik szintén az EEG adatok elemzésére szolgálnak. Ezek a hálózatok különböző agyi jellemzők detektálására alkalmasak, amivel bővítik az EEGNet által nyújtott lehetőségeket. Ezenkívül a BrainNetCNN [34], amely az agyhálózatok elemzésére specializálódott, valamint a Temporal Convolutional Networks (TCN) [35], amelyek a hosszú távú függőségek tanulásában nyújtanak előnyöket, további eszközöket biztosítanak az EEG adatok sokrétű elemzéséhez. [9]

A P300 jel viszonylagos egyszerűsége miatt nincs szükség komplex neurális hálóra. Az

EEGNet egy általános célú hálózat, amely ezen a területen már bizonyított hatékonysággal rendelkezik. [9] Ezzel szemben a többi modell elsősorban motoros paradigmákra lett tervezve, ami kevésbé teszi őket alkalmassá a P300 jel detektálására.

3.5. Unicorn Hybrid Black

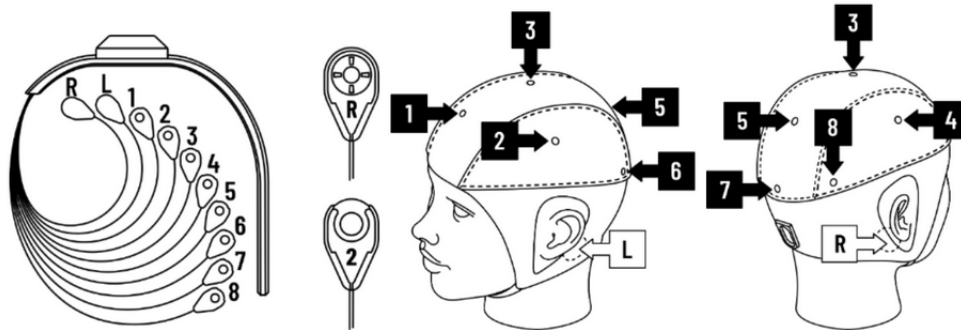


3-6. ábra. A g.tec Unicorn Hybrid Black fejhallgató. [36]

Konzulensemnek köszönhetően hozzáférhettem egy a g.tec nevű cég által fejlesztett Unicorn Hybrid Black (3-6-os ábra) nevű (továbbiakban Unicorn) EEG jeleket rögzíteni tudó fejhallgatóhoz. Az Unicorn egy korszerű, fogyasztói szintű bioszignál erősítő készlet, amelyet kifejezetten arra terveztek, hogy széles körű alkalmazási lehetőségeket nyújtson a fejlesztők, művészek és egyéb kreatív szakemberek számára. [37] Ez a rendszer lehetővé teszi, hogy a felhasználók az emberi testből származó jeleket integrálják különböző projektekbe, így az EEG jelek egyszerű megjelenítésétől kezdve az ezekre reagáló eszközök tervezésén és vezérlésén át, művészeti installációkkal való interaktív kapcsolatok kialakításáig.

Az Unicorn a Unicorn Hybrid EEG elektródák használatával rögzíti a nyolc csatornás EEG jeleket. Az eszköz kialakítása a standard 10-20-as elrendezést követi az elektródák pozicionálásában, ami elősegíti az EEG jelek pontos és megbízható rögzítését. A szoftveres támogatásról az Unicorn Suite gondoskodik, amely egy integrált szoftverkörnyezetet

biztosít, beleértve az önálló alkalmazásokat és a programozási felületeket, amelyek lehetővé teszik a felhasználók számára az adatok rögzítését, feldolgozását és azonosított BCI paradigmák alkalmazását.



3-7. ábra. Az Unicorn fejhallgató elektródái és elhelyezkedésük az eszközön. [37]

Az 3-7-es ábrán látható az Unicorn elektródáinak elhelyezkedése és elnevezése. Minden csipesz csatlakozó alján szám vagy betű található, ami a helyes pozíciót jelöli. A sapka előre meghatározott elektróda pozíciói Fz, C3, Cz, C4, Pz, PO7, Oz és PO8 a korábban említett 10/20 rendszer szerint. Az L (bal) és R (jobb) jelölésekkel ellátott jelölések az Unicorn Sticky elektródákhoz tartoznak. Ahogy az ábrán is látható ezek a speciális elektródák a bal és jobb oldali mastoidhoz kell felragasztani, azért, hogy referenciaként szolgáljanak a többi csatornához.

3.6. Felhasználói interfész

A grafikus felhasználói interfész (a továbbiakban GUI) tervezésekor több Python alapú grafikus könyvtár közül lehetséges a választás, mint például a Tkinter [38], Pygame [39], és PsychoPy [40], melyek mindegyike eltérő előnyöket és hátrányokat kínál a grafikus alkalmazások fejlesztéséhez.

Tkinter egy Python alapú GUI könyvtár, amely beépített támogatást nyújt, nem igényel külön telepítést, és egyszerű szintaxisa révén kiválóan alkalmas alapvető grafikus felhasználói felületek elkészítésére. Széleskörű dokumentációja révén könnyen elsajátítható. Azonban a Tkinter korlátozott funkcionalitása és kissé elavult megjelenése kevésbé teszi vonzóvá komplex vagy nagyon interaktív UI-k fejlesztéséhez. [38]

Pygame erős multimédia támogatást nyújt, ideális játékok és multimédia alkalmazások fejlesztésére. Kiválóan kezeli a felhasználói interakciókat, mint az egér és billentyűzet

eseményeket, és gyors, hatékony teljesítményt nyújt komplex grafikák és animációk kezelésében. [39]

PsychoPy, kifejezetten pszichológiai, neurológiai kísérletekhez és viselkedéstudományi vizsgálatokhoz fejlesztett könyvtár, amely időzített vizuális és audio stimulációkat tesz lehetővé. Könnyen integrálható más tudományos és adatkezelő könyvtárakkal, mint a numpy vagy az MNE, támogatja az adatgyűjtést és -elemzést, amely különösen hasznos kísérleti környezetben. Specializált jellege és a magas tanulási görbe azonban korlátozhatja az általános célú alkalmazások fejlesztésében. [40]

3.7. A megoldási módszer kiválasztása

Irodalomkutatásom során nem sikerült olyan komplett megoldást, leírást találnom ami az előttem álló feladatot maradéktalanul meg tudta volna valósítani. Ebből kifolyólag az szakdolgozatomat az alapoktól felépítve, minden komponenst egyedileg kell majd elkészítenem. Az előttem lévő lehetőségek közül ezért a döntésemet alapvetően befolyásolta, hogy melyik komponenssel illetve programozási nyelvvel van már magabiztos tapasztalatom.

Az EEG kísérlet kialakításához szükséges összetett alkalmazás tervezésénél elengedhetetlen, hogy a választott programozási nyelv megfeleljen az EEG jelek gyors és hatékony feldolgozásának igényeihez. Ezért a Python programozási nyelv erre a kihívásra kifejezetten alkalmas. A Python egy magas szintű nyelv, amelynek egyszerű szintaxisa és gazdag könyvtár támogatása lehetővé teszi, hogy kevés előkészítő munkával, mégis hatékonyan létrehozzunk összetett szoftvermegoldásokat.

A felhasználói felület fejlesztéséhez több Python-alapú könyvtár is rendelkezésre áll. A PsychoPy, amely kimondottan pszichológiai és neurológiai kísérletekhez készült, kiválóan alkalmas időzített vizuális és audio stimulációk elvégzésére, ami létfontosságú a P300 jelek kiváltásához szükséges kísérletekben. Azonban a tervezési fázis során elkészített demo alkalmazásokkal szerzett tapasztalatok miatt mégis a Pygame választása felé sarkalltak. Hiába az agyhullámok vizsgálatához szükséges kísérletekre lett felkészítve a PsychoPy, a fejlesztés a csomag használatával bosszantóan lassúnak és frusztrálónak bizonyult. Ennek eredményeképpen a döntésem végső soron a Pygame-re esett, hiszen az alkalmas interaktív, multimédia alkalmazások hatékony fejlesztésére.

A P300 jelek azonosítására képes programnak szükségszerűen fel kell tudnia dolgoznia a beérkezett EEG adatokat. A Python nyelven rendelkezésre álló MNE könyvtár kiemelkedően hasznos az ilyen típusú neurofiziológiai adatok vizsgálatát, mivel kifejezetten az

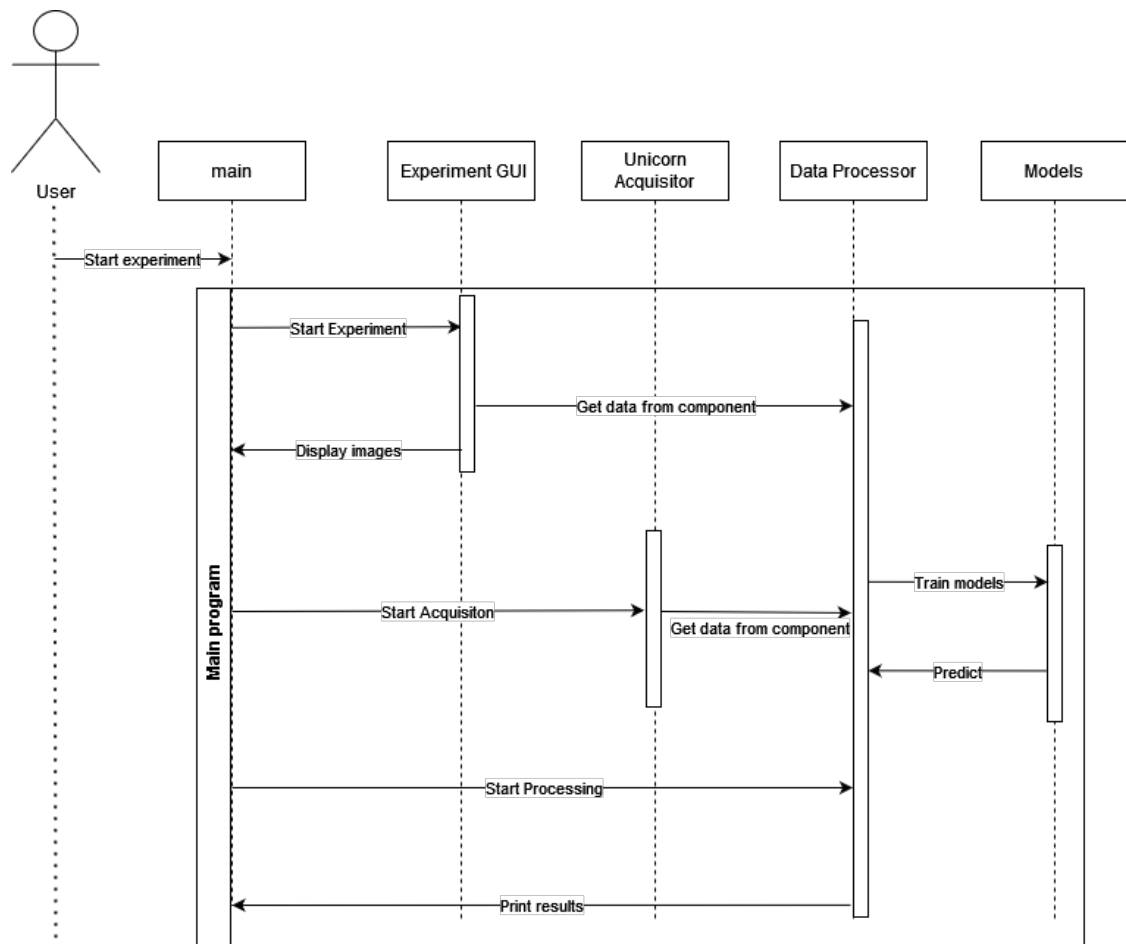
EEG jelek elemzésére lett kifejlesztve. Emellett a numpy könyvtár biztosítja az adatok gyors és hatékony kezelését, lehetővé téve a nagy mennyiségű adat komplex, matematikai manipulációját.

Az Unicorn Hybrid eszközzel való kommunikáció is támogatott mind C#, mind Python nyelven, de a Python nyelv kiválasztása a fent említett okokból előnyösebb a projekt szempontjából. Az egyesített megoldás, amely magában foglalja a Python nyelvet, valamint az MNE, numpy és Pygame könyvtárakat, lehetővé teszi a P300 jelek pontos és hatékony detektálását és elemzését.

4. Követelmény specifikáció

Célomnak egy olyan BCI rendszer kialakítását tűztem ki, amely képes a P300-as ERP detektálására valós időben. A rendszer lehetővé teszi, hogy a felhasználók a vizuális stimulációk révén előidézzék a P300 jelet, amit a képernyővel való interakcióhoz lehet majd felhasználni. Az átláthatóság miatt a funkciók és modulok interakciójáról elkészítettem egy szekvenciadiagrammot amit a 4-8-as ábrán lehet látni.

A rendszer alkalmas kell legyen különböző vizuális stimulációk, azaz képek megjelenítésére, amelyek véletlenszerű frekvenciált váltakozása előidézi a P300 jelet (4-8-as ábrán az Unicorn Experiment GUI). A stimulációk pontos időzítése kiemelten fontos, ezért a lehető legpontosabb módon kell tudni azokat rögzíteni. Ez azért szükséges, hogy egyértelműen azonosítható legyen a P300 jelek kiváltásának ideje. Nélkülözhetetlen tehát, hogy a rendszer komponensei szinkronban tudjanak egymás mellett működni.



4-8. ábra. Az applikáció szekvenciadiagramja bemutatja a program működését.

A rendszert fel kell készíteni, hogy képes legyen kommunikálni az EEG eszközökkel (a

4-8-as ábrán az Unicorn Acquisitor modul). Az Unicornról érkező adatokat időbélyeggel ellátva kell tudni eltárolni a további elemzésekhez. A rendszernek periodikusan, meghatározott időközönként kell képesnek lennie az EEG adatok lekérdezésére és továbbítására.

Az EEG adatokat megfelelően kell előkészíteni az elemzéshez (4-8-as ábrán az Unicorn Processor). Az adatokból az artefaktumok eltávolítása mellett sávszűrést is alkalmazni kell a zaj csökkentése érdekében. A feldolgozott adatokat olyan formátumra kell hozni, ami kompatibilis az MNE analízis eszközökkel. Az előkészített adatokat az egyes események alapján epochákra kell bontani az MNE csomagban található Epochs objektumok segítségével.

Végül a rendszernek fel kell tanítania egy modellt a P300 jelek felismerésére (4-8-as ábrán a Models rész), amelyet aztán valós idejű adatokon is tesztelni kell. Az EEGNet neurális háló és az LDA használata a P300 jellemzők felismerésére és osztályozására ideális, mivel ezek a modellek különösen alkalmasak az EEG adatok jellemzőinek hatékony felismerésére. A feltanított modellt valós idejű adatokon kell tesztelni, az eredményeket tárolni és a felhasználók számára visszajelzést adni a prediktált P300 jelek száma alapján.

Ezek a követelmények biztosítják, hogy a rendszer hatékonyan és pontosan legyen képes kezelni és elemezni a P300 eseményhez kötött potenciálokat, támogatva ezzel a neurológiai kutatásokat és a klinikai diagnosztikát.

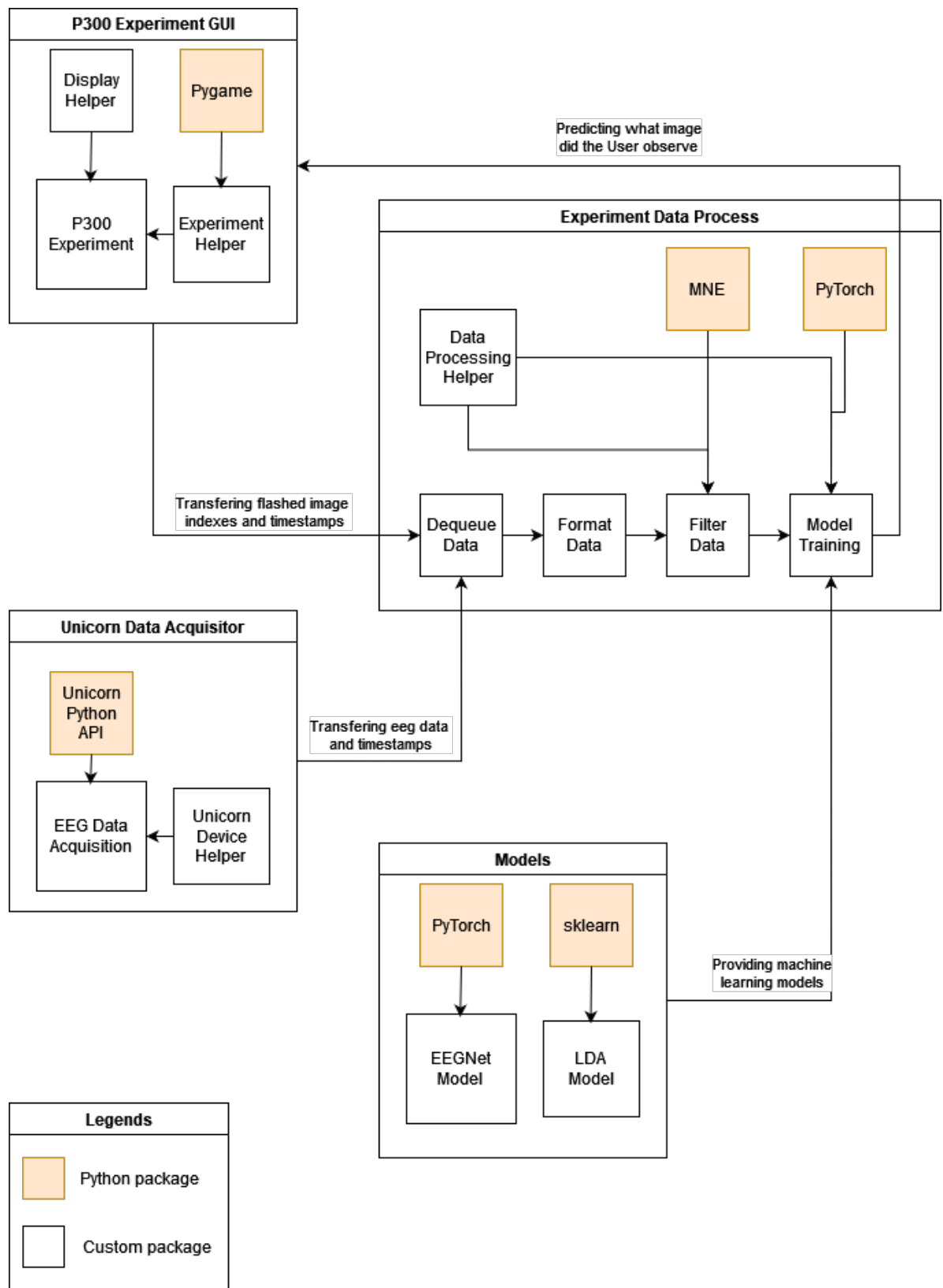
5. Rendszerterv

A korábbi fejezetekben ismertettem a BCI-t, szót ejtettem az invazív és non-invazív módszerekről, bővebben értekeztem az EEG jelekről, illetve a hozzá tartozó jelentősebb paradigmákról. A kutatásom sikeres végrehajtásához, a tanulmányozott szakirodalomnak köszönhetően kaptam egy átfogó képet a BCI-k illetve az EEG alapú jelfeldolgozások jelenlegi állapotáról.

Világossá vált számomra, hogy a non-invazív módszerek is képesek jó eredményeket elérni, amiket a gépi tanulás segítségével még inkább tudunk pontosítani és gyorsítani. Ahogy arra már korábban is kitértem az EEG mellett döntöttem, mert ez a fajta mérés nem igényel sok előkészületet, biztonságosabb, valamint költség- illetve időhatékonyabb.

A rendszer egy integrált alkalmazást képez, amely négy kulcsfontosságú összetevőből áll: egy a kísérlet megjelenítésért felelő grafikus interfész, egy modul amely adatokat gyűjt az Unicorn eszközből, egy az előző két komponensből összegyűjtött adatokat feldolgozó programrész és végül a gépi tanulást megvalósító modellek. Az alkalmazás alapvető célja az EEG jelek valós idejű feldolgozása és analízise, melyhez a korábban kiválasztott gépi tanulási modelleket fogom alkalmazni. Ezt a rendszertervet az 5-9-es ábrán szemléltetem vizuálisan és következő bekezdésekben fejtem ki a részek viselkedését.

A felhasználói interfész elsődleges szerepe, hogy dinamikusan megjelenítse a vizuális stimulációk sorozatát. A felület véletlenszerű időközönként képeket jelenít meg, amelyeket az EEG eszköz rögzít. A UI folyamatosan naplózza majd az egyes képek indexét és azok felvillanásának nanoszekundumra pontos idejét, amely információk elengedhetetlenek a későbbi adatelemzés folyamatok során. A modellek tanítási időszakában szükséges lesz, hogy véletlenszerűen kiválasszunk egy képet a képernyőn láthatóak közül. Ez azért is kritikus, mert valahogyan meg kell majd tudnunk határozni, hogy melyik is a „cél” kép amit a felhasználó éppen néz, annak érdekében, hogy meg tudjuk állapítani, hogy az adott intervallumban jelent volt-e a P300-as jel.



5-9. ábra. Az applikáció rendszerterve

A rendszer második kulcskomponense az adatgyűjtő kommunikációs interfész, amely felelős az Unicorn eszközzel történő adatcsereért Bluetooth kapcsolaton keresztül. Az eszköz működését egy licenz alapú Python API szabályozza, amely lehetőséget biztosít arra, hogy meghatározott frekvencián, egész pontosan 250 Hz-en, lekérje az EEG adatokat a nyolc csatornán keresztül. Az adatgyűjtés folyamata során elengedhetetlen, hogy minden egyes adatpontot pontos időbélyeggel lássunk el (azaz az adatok pontos beérkezési idejét rögzítsük), amely kulcsfontosságú az időfüggő elemzések szempontjából.

A gyűjtött adatok valós időben történő időbélyegzése és a folyamatos adatáramlás menedzselése érdekében az említett Python API-t kell alkalmaznunk. Ez a megközelítés lehetővé teszi, hogy az adatgyűjtési folyamatot könnyedén elindíthassuk az eszközön, valamint folyamatosan fogadhassuk és rögzíthessük az adatokat, miközben a rendszer többi része is zavartalanul végezheti a saját feladatait. Az így kialakított rendszer maximálisan kihasználja az Unicorn eszköz adottságait és biztosítja az adatok integritását az elemzési folyamat minden szakaszában.

A rendszer harmadik modulja az érkező EEG adatok feldolgozásáért és elemzéséért felelős. Ez a komponens meghatározott időközönként veszi át és dolgozza fel azokat az adatokat, amelyeket a felhasználói interfész és az EEG adatgyűjtésért és továbbításáért felelős előző két modul szolgáltat. A feldolgozás során a rendszernek képesnek kell lennie arra, hogy speciális szűrőalgoritmusok segítségével kiszűrje azokat az artefaktumokat, amelyeket például a szemmozgások, izomműködések vagy környezeti zajok okoznak.

Ebben a szegmensben kaptak helyet az EEG hullámok tisztításáért felelős szűrők. A szűrők nélkülözhetetlen eszközök az EEG adatok feldolgozásában, mivel lényegesek a jelminőség javításában és a nem kívánt zajok kiszűrésében. A leggyakrabban alkalmazott szűrőtípusok között szerepelnek az aluláteresztő szűrők, amelyek a magas frekvenciájú zajokat szűrik ki, és csak a meghatározott vágási frekvencia alatti jeleket engedik át. A felüláteresztő szűrők az alacsony frekvenciájú zajokat távolítják el, csak a meghatározott vágási frekvencia feletti jeleket engedve át. A sávszűrők az alacsony- és magasfrekvenciás zajok kombinált kiszűrésére szolgálnak, míg a metszetszűrők specifikus, keskeny frekvenciatartományok, mint például a hálózati zaj (általában 50 vagy 60 Hz) kiszűrésére alkalmazhatók. Ezeknek a szűrőknek a kombinációját fogom majd alkalmazni az EEG adatok tisztítására.

A rendszertervem egy integrált alkalmazás, amely négy alapvető modult foglal magában: egy grafikus felhasználói interfészt, egy EEG adatgyűjtő modult, egy az adatok feldolgozásáért és tisztításáért felelős komponenst és egy gépi tanulási modult. A felhasználói interfész a vizuális stimulációk dinamikus megjelenítéséért felel, míg az adatkezelő mo-

dul strukturáltan gyűjti és tárolja az Unicorn eszköztől kapott EEG adatokat. Az adatgyűjtő modul fogadja, feldolgozza és átalakítja az előző program részekről kapott adatokat. A gépi tanulási modul pedig a gyűjtött adatok alapján végzi az osztályozási és mintafelismerési feladatokat. A rendszer kiemelten kezeli az adatok valós idejű időbélyegzését és a kommunikációs interfészen keresztül történő adatátvitelt, biztosítva az adatgyűjtés és -feldolgozás zökkenőmentes működését. Az EEG artefaktek kiszűrésére számos szűrőtechnika kerül alkalmazásra, melyek jelentősen javítják a jelminőséget és a mérések pontosságát. A rendszer moduláris felépítése és a legkorszerűbb technológiák használata teszi lehetővé, hogy a program hatékonyan működjön a gépi tanulás segítségével, kiaknázva az EEG adatok elemzésében rejlő lehetőségeket.

6. Megvalósítás

6.1. Fejlesztői környezet

Szakedolgozatom forrásállományának megvalósítását a megfelelő fejlesztői környezet kiválasztásával kezdtem. Egyetemi tanulmányaim és saját projektjeimnek köszönhetően magabiztosággal választottam a Visual Studio Code (VS Code) környezetet, ami egy ingyenes, nyílt forráskódú szerkesztő és rendkívül testreszabható. Támogatja a Python fejlesztés számos aspektusát, beleértve az intelligens kódkiegészítést, a kódhibák javítását, valamint a refaktorálást. Az integrált hibakereső segíti a Python kód lépésenkénti végrehajtását, a változók megfigyelését és a töréspontok kezelését, miközben támogatja a legnépszerűbb Python teszt keretrendszereket is. A VS Code lehetővé teszi az interaktív kódfejlesztést, amely különösen előnyös az EEG jelek feldolgozásánál, ahol az adatvizualizáció és gyors eredményellenőrzés fontos. Ezek a jellemzők teszik a VS Code-ot ideális választássá minden Python fejlesztő számára, különösen azok számára, akik az EEG jelek feldolgozásával foglalkoznak.

6.2. Verziókövetés

A fejlesztés során gyakran kellett váltogatnom a laptopom és a személyi számítógépem között így hozva létre egy körülményesebb fejlesztési metodika. Emiatt meghatározó volt a projektem verzió követése, aminél hasznosíthattam a tanulmányaim és a munkám során felszedett Git tudást. Nehezítésként jelentkezett, hogy a fejlesztéshez használt eszközeimen eltérő verziójú Python környezeteket használtak. Ennek orvoslására létrehoztam a Pyenv [41] és Pipenv [42] csomagok segítségével egy virtuális környezetet. A Pyenv és a Pipenv két hasznos eszköz Python fejlesztők számára, melyek segítenek a fejlesztési környezet kezelésében és a projekt függőségek szabályozásában.

A Pyenv egy sokoldalú eszköz, amely lehetővé teszi több Python verzió egyidejű kezelését egy adott rendszeren. Ez különösen hasznos, amikor új nyelvi funkciókat szeretnénk tesztelni, vagy különböző Python verziókat igénylő projekteken dolgozunk. A Pyenv segítségével egyszerűen telepíthetjük a Python legújabb fejlesztési verzióit, váltogathatunk a különböző verziók között, és virtuális környezeteket hozhatunk létre, valamint aktiválhatunk automatikusan. [41]

A Pipenv, amely egy integrált eszköz a Python projekt csomagkezelésére és virtuális környezet kezelésére, automatikusan kezeli a csomagokat a Pipfile segítségével, amikor új csomagokat telepítünk vagy eltávolítunk. A Pipenv egyesíti a Pip és Virtualenv funkcióit, így nem szükséges ezeket külön használni, továbbá kiküszöböli a requirements.txt

fájlt, helyette a Pipfile-t használja a függőségek nyilvántartására és a Pipfile.lock-ot az alkalmazás buildelésére. A Pipenv támogatja a virtuális környezetek gyors és egyszerű beállítását, és lehetővé teszi a verzió nélküli csomagtelepítést, ami csökkenti az ütközéseket a különböző függőség-verziók használata esetén. [42]

Ezen eszközök szabványosítása fontos, mivel segít áthidalni azokat a kihívásokat, amelyek akkor jelentkeznek, amikor a fejlesztési eszközök közötti eltérések miatt a környezetet újra létre kell hozni. Ezen kívül csökkenti a függőségi problémákat, amelyek változásokat okozhatnak a csomagverziókban és potenciálisan a modell teljesítményében.

6.3. Konfiguráció

A programom átláthatóságának és karbantarthatóságának javítása érdekében egy külön konfigurációs fájlt hoztam létre, amelybe kiemeltem a futáshoz szükséges konstans értékeket. Ez a megközelítés elősegíti a kód modularitását és egyszerűsíti a változások kezelését anélkül, hogy közvetlenül a programkódhoz kellene nyúlni. A konfigurációs fájl tartalmazza a kísérlethez szükséges képek elérési útvonalait, a vizuális stimulációk frekvenciáját, a megjelenítendő szövegek színbeállításait, valamint a jelzésfeldolgozás során alkalmazott szűrési paramétereket. Emellett definiálja azokat a csatornákat is, amelyeket az adatfeldolgozás során az epochálásnál kell használni, továbbá a gépi tanulási modellek hiperparaméterei is itt kaptak helyet.

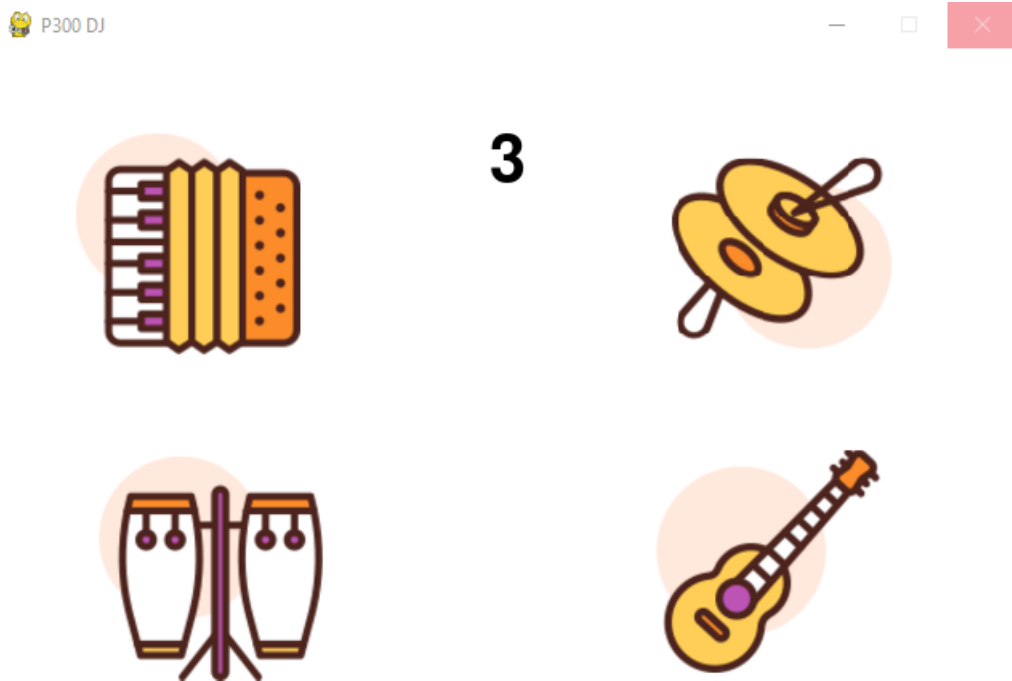
Ez a struktúra lehetővé teszi, hogy a kísérleti beállítások gyorsan és könnyedén módosíthatóak legyenek, így biztosítva a kísérletek közötti konzisztenciát és a különböző beállítások hatékony tesztelését. A konfigurációs fájl használata növeli a projekt átláthatóságát, és segít abban, hogy a kódbázis rendezett és jól strukturált maradjon, ami mérvadó a hibák minimalizálása és a fejlesztési folyamat hatékonyságának maximalizálása szempontjából.

6.4. A kísérlet grafikus interfésze

Miután sikeresen létrehoztam a fejlesztéshez szükséges virtuális környezetet, hozzáfogtam a grafikus felhasználói interfész kialakításához. Mint korábban említettem, a Pygame csomagot választottam az experimentális felület megteremtésére. A kialakítás során igyekeztem a „Clean Code” és a SOLID elvek [43] betartására. Létrehoztam egy ExperimentGUI nevű osztályt, amely a P300 kísérlet során használt vizuális elemek megjelenítéséért és a GUI futásához szükséges segítő osztályok összekapcsolásáért felelt. Az ExperimentGUI osztályban vezetem az aktuálisan kiválasztott képek indexeit, és itt rögzítettem az egyes képek megjelenésének időpontját nanoszekundumos pontosságú időbélyeggel

ellátva. A modellek betanításának időszakában szükségem volt arra, hogy nyilvántartsam az éppen véletlenszerűen kiválasztott kép indexét, valamint azt az időintervallumot, amíg a kiválasztott kép el nem érte a kívánt mennyiségű képvillanás.

Az időbélyegek minnél pontosabb létrehozása miatt a Python `time` csomag `time_ns()` függvényét használtam. A `time_ns()` az úgynevezett epoch óta eltelt időt adja meg. Az epoch, amely az időszámítás kezdetét jelenti, platformtól függően változik; a legtöbb Windows és Unix rendszeren 1970. január 1., 00:00:00 (UTC) kezdődik, és ez alatt a szökőmásodpercek nem számítanak bele. [44]



6-10. ábra. Az alkalmazás grafikus kinézetének megvalósítása. A képernyőn négy rajzolt hangszer jelenik meg egymástól egyenlő távolságra. Ezek a képek véletlenszerű sorrendben villannak fel. Az ábra tetején látható szám, a tanulási szakaszban indikálja a felhasználó számára, hogy melyik képre kell fókuszálnia.

A `pygame_setup.py` segéd fájl felelt a Pygame instancia konfigurálásáért és inicializálásáért. Ebben a fájlban többek között a képernyő méretét, színét és a megjelenítendő szöveg méretét valamint annak betűtípusát lehet beállítani. Az `experiment_helper.py` fájlba szerveztem ki a képek megjelenítésért, „villogtatásáért” és a szövegek ábrázolásáért felelős kódrészeket.

Az alkalmazás kinézete a 6-10-es ábrán látható módon lett kialakítva. Grafikának hangszereket választottam, mert az egyik tovább fejlesztési célom az, hogy egy zeneszerkesztő programot készítek el, amit a BCI segítségével lehet irányítani. A képeket egymástól egyenlő távolságokra jelenítettem meg és a képernyő közepétől kicsit feljebb foglal helyet a tanítás során használt index megjelenítés. Az éles működésnél a feltanított modell predikciója után itt jelenik meg annak a képnek az indexe amire a felhasználó fókuszál.

6.5. A kísérlet leírása

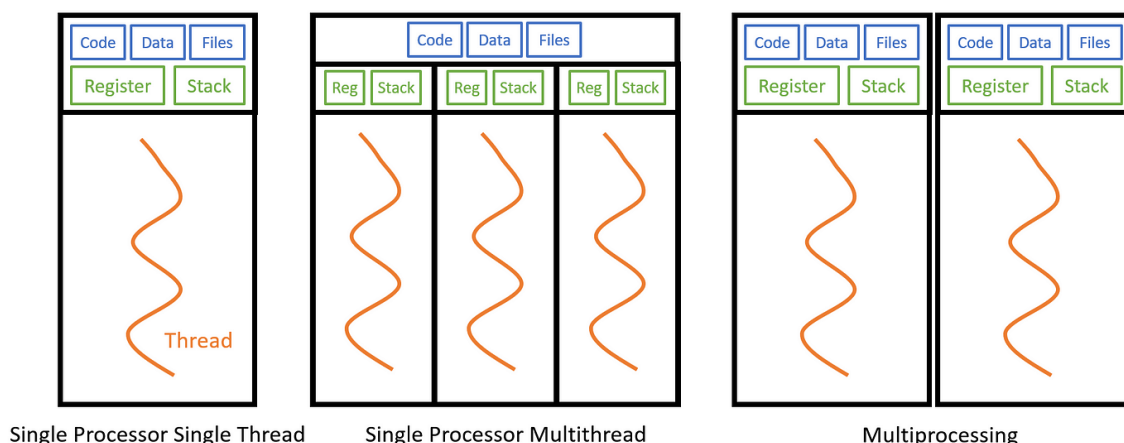
A kísérlet során használt program grafikus felületén négy hangszer jelenik meg egy fehér háttéren, amelyek véletlenszerű sorrendben villannak fel. A tanítási adatok gyűjtése során a képernyőn egy véletlenszerűen kiválasztott kép indexe jelenik meg a négy közül. Ezt követően egy újabb kép indexe jelenik meg, egészen addig, amíg minden kép indexe meg nem jelent a kijelzőn.

A tanulási folyamat addig tart, amíg minden, a képernyő közepén jelzett indexű kép fel nem villan 30-szor, mivel az Unicorn eszköz ennyi próba után tud 90 százalék körüli pontosságot elérni. [45] A képek felvillanása között 400 milliszekundumos szünetek vannak beiktatva. Ezen felül a kódban biztosítom, hogy egy kép ne villan hasson fel egymás után kétszer.

6.6. Párhuzamos programozás implementálása

Ahhoz, hogy minél akkurátusabban tudjuk észlelni a P300-as jeleket az agyhullámokban, kardinális, hogy a program futása a lehető leggyorsabb legyen. Ebből következik, hogy a program valamennyi komponensének párhuzamosan kell tudni operálnia. Alapvetően a Python nyelv nem a párhuzamos alkalmazások készítésére lett kifejlesztve, azonban egy kis hatékonyság beáldozásáért cserébe elkészíthető egy ilyen program. Ezt a kihívást két féleképpen közelíthettem meg; többszálás vagy többfeldolgozóssá keltet alakítanom a programomat.

A többszálásítás (az 6-11-es ábrán a középső kép) különösen hatékony IO-korlátozott feladatoknál, mivel lehetővé teszi az egyidejű feldolgozást. Ezzel szemben a többfeldolgozás (az 6-11-es ábrán az utolsó kép) előnyös CPU-igényes műveleteknél, például bonyolult számítási feladatok esetén, mert kihasználja a többmagos processzorok adta előnyöket, így gyorsítva a feldolgozást. [46]



6-11. ábra. A képen három eltérő architektúra látható. Az elsőben egy processzoron fut egy szál, a másodikon egy többszálú architektúra ami több szálat használ egyetlen processzoron belül az utolsó képen pedig a többfeldolgozós architektúra kerül ábrázolásra, amiben több processzor futtat párhuzamosan több szálat. [47]

Választásom először a többfeldolgozás implementálására esett. Az egyidejűleg futó folyamatok közötti adat kommunikációhoz egy párhuzamos programozásnál használatos multiprocessing csomagban található Queue típusú objektumot használtam. Ez a Queue objektum a standard Python csomagban található queue.Queue objektumra hasonlít. Egyik előnye, hogy használatával a különböző szálak egyidejűleg és konfliktusmentesen kezelni tudják az adatokat, így megelőzve az adatvesztést vagy a hozzáférési hibákat, amelyek több párhuzamosan futó folyamat esetén fordulhatnak elő. [48]

A programom ebben a kezdeti fázisban azonban nem várt hibára futott. A model tanítása során az adatokat folyamatos Queue objektumba történő tárolása miatt a raktározásra használt unix socket megtelt, ezért blokkolta az adott szál futását. Ez azért jelentett problémát, mert így nem tudott megszakítás nélkül leállni a program. A megoldást a többszálúítás féle megközelítés jelentette és így már zökkenőmentesen és saját szálaikon tudtak futni a grafikus interfészt, adatgyűjtést és az adat feldolgozást megvalósító kódrészek. Ennek az alfejezetnek lezárása képpen megemlíteném, hogy a multiprocessing típusú Queue objektumot lecseréltem a Pythonhoz tartozó threading modulban megtalálható Queue objektumra.

6.7. Adatok gyűjtése az Unicornnal

Az Unicorn-nal való adatgyűjtés esszenciális része a Unicorn.py nevű Python nyelvre megírt API, amit egy a g.tec-től megvásárolt licensszel lehet aktiválni. A csomag importálásának megtörténte után lehetőségünk van a bekapcsolt Unicorn eszközhöz kap-

csolódni. Az Unicorn UDP interfész az adatcsomagokat 250 Hz-es mintavételi frekvenciával küldi. Ezek az adatcsomagok 17 csatornáról származó adatokat tartalmaznak az eszközből (8 EEG csatorna, 3 gyorsulásmérő, 3 giroszkóp csatorna, akkumulátorszint, számláló, érvényesítés) amelyek lebegőpontos formátumúak. Ennek megfelelően minden egyes adatcsomag 68 bájtból áll, amit lebegőpontos tömbbé kell konvertálni a csomagok beérkezése után.

```
def session(self):
    self.session_buffer = []

    try:
        if self.unicorn_device is not None:
            self.unicorn_device.StartAcquisition(unicorn_experiment_config.test_signal_enabled)
            self.stop_session = False

            while not self.stop_session:
                start_timestamp = time.time_ns()

                self.unicorn_device.GetData(unicorn_experiment_config.frame_length, self.receive_buffer, self.receive_buffer_buffer_length)

                data = np.frombuffer(self.receive_buffer, dtype=np.float32, count=self.number_of_acquired_channels * unicorn_experiment_config.frame_length)
                data = np.reshape(data, (unicorn_experiment_config.frame_length, self.number_of_acquired_channels))
                EEG_data = data[:, unicorn_experiment_config.EEG_channels]
                actual_data_received = EEG_data.shape[0]

                self.session_buffer.append(np.array(EEG_data.T))
                self.acquired_data_timestamps.append(log_helper.get_time_stamp_stream(start_timestamp, actual_data_received))
            except UnicornPy.DeviceException as e:
                print(e)
            except Exception as e:
                print("An unknown error occurred. %s" %e)
```

6-12. ábra. Az adatok kigyűjtése a Unicorn eszközről

Az adatok gyűjtésére létrehoztam az UnicornDataAcquisitor osztályt, amit támogat egy unicorn_device.helper.py nevű fájl, aminek a feladata az elérhető Unicorn eszköz detektálása. A számítógéphez csatlakoztatott Unicorn eszköz StartAcquisition metódusával lehetséges a rögzített agyhullámok tényleges lekérése. Az 6-12-es ábrán látható session metódusban az összegyűjtött EEG adatokat ezután leszűrtem a 8 EEG csatornára, mivel ezek azok a csatornák amelyek a P300 jel detektálásához nélkülözhetetlenek. Az EEG adatokat az akvizíció után egy pufferbe helyezem, majd a megfelelő formátumra alakítom azokat és eltárolom őket a session.bufferbe. A session metódus végén pedig nanoszekundumos pontosságú időbélyeggel látom el az eszköztől gyűjtött adatokat.

Az összegyűjtött adatok naplózását végző függvény get_time_stamp_stream névre hallgat és a 6-13-es ábrán látható. A függvény az adatmennyiség függvényében generál időbélyegeket egy kezdő időponttól kiindulva (ez a 6-12-es ábrán start.timestamp néven látható változóban eltárolva), célja minden egyes adatpont számára pontos időbélyegek előállítás. Első lépésként inicializál egy tömböt nulla értékekkel, aminek mérete megegyezik a fogadott adatok mennyiségével. Ezt követően kiszámítja a mintavételek közti időt milliszekundumban a konfigurációs fájlban meghatározott mintavételi ráta alapján, amely meghatározza a mintavételi intervallumot. A folyamat során a függvény iterál minden adatponton, ahol kiszámítja az egyes adatpontok közötti időkülönbséget, majd ezt

```

def get_time_stamp_stream(start_timestamp, actual_data_received):
    time_stamps = np.zeros(actual_data_received)
    for i in range(actual_data_received):
        time_between_two_samples = 1000 / unicorn_experiment_config.sample_rate
        delay_ms = time_between_two_samples * (actual_data_received - i)
        delay_ns = delay_ms * 1000000
        time_stamps[i] = (
            start_timestamp - delay_ns
        )
    return time_stamps

```

6-13. ábra. Az kigyűjtött adatokat időbélyegző algoritmus

nanoszekundumra konvertálja, így az időbélyegek egységes mértékegységben kerülnek rögzítésre. Az adatgyűjtés végén a függvény visszaadja a teljes időbélyeg tömböt, amely tartalmazza minden egyes adatpont pontos időpontját, ezáltal kritikus alapot nyújtva a későbbi analízisekhez és az adatok értelmezéséhez.

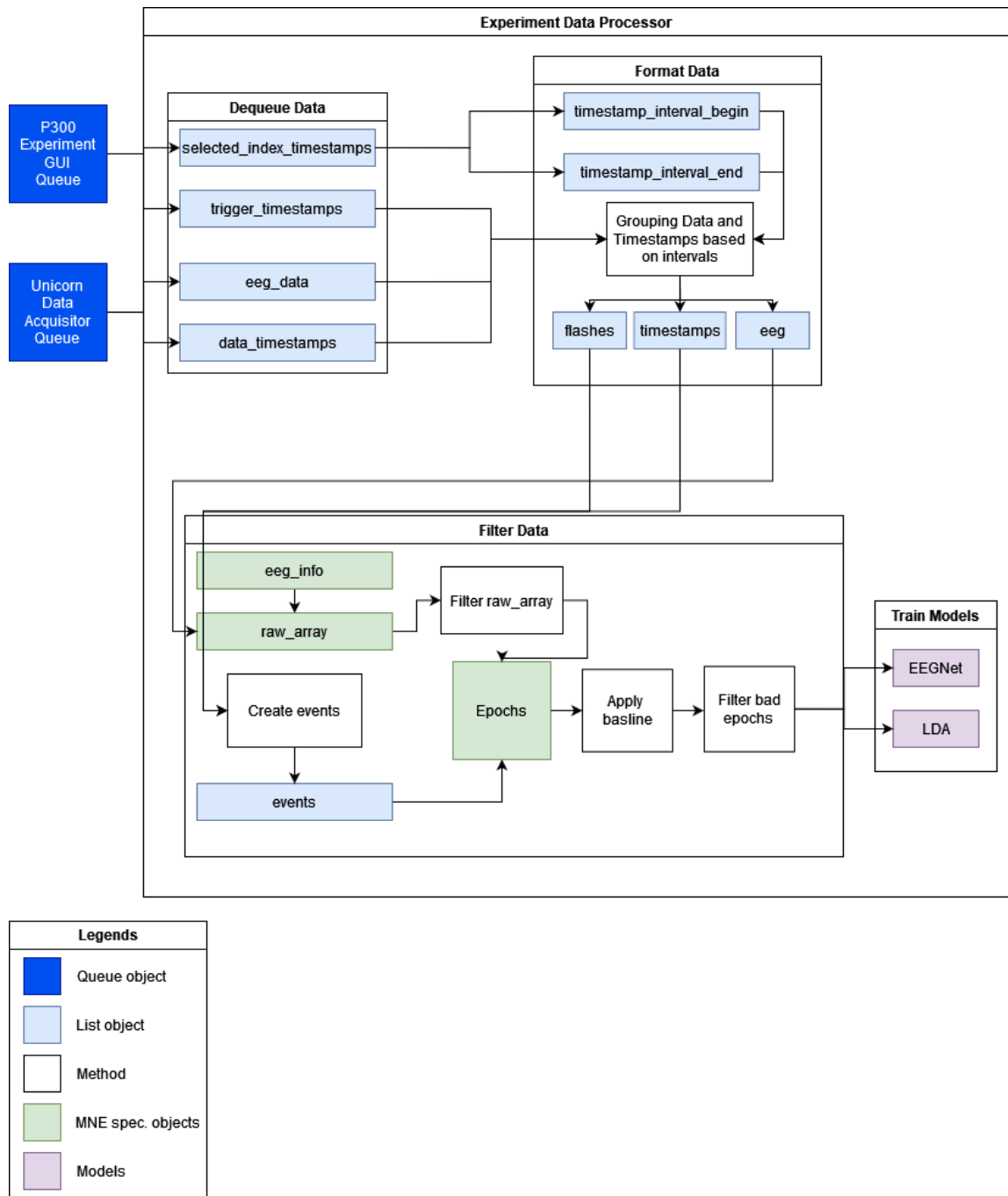
A külön tárolt formázott EEG adatok és időbélyegek a kísérlet lefutása után egy Queue objektumban kerültek tárolásra, amely lehetővé teszi, hogy a több szálon futó folyamatok biztonságosan és hatékonyan férjenek hozzá a megosztott adathalmazhoz.

6.8. Adatok feldolgozása

Az ExperimentDataProcessor osztály fogja össze az adatok feldolgozását és a modellek tanítását. Az osztály működését a 6-14-es ábra mutatja be. A modul úgy lett elkészítve, hogy az ExperimentDataProcessor objektum inicializálásakor egy osztályszintű training_mode elnevezésű kapcsoló segítségével tudunk váltani az „éles” és tanulási módozatok között.

A tanulási mód kiválasztása esetén a program futása a következő lépéssel folytatódik. A párhuzamosan futó GUI és Unicorn EEG adatok gyűjtését végző szálakon használt Queue objektumokból külön osztályszintű listákba történik az információ kiemelése. Ez pontosan négy listát jelent. Az első lista az eeg_data nevet kapta és az Unicorn eszközzel gyűjtött EEG adatokat tartalmazza. A második neve data_timestamps és itt található az EEG adatok beérkezésének pontos időpontjai. A harmadikban vannak a GUI Queue-ban továbbított a tanítási fázisban a felhasználó figyelmének megragadásához használt kép indexe, és az intervallum amíg azt a képet kellett figyelnie. Ennek a listának a neve: selected_index_timestamps. Végül az utolsó lista a trigger_timestamps, amiben a véletlenszerűen felvillant képek indexei és a villanás időpontja lett letárolva.

A külön válogatott adatokat ezután olyan formátumúra hozzuk, hogy az MNE csomag-



6-14. ábra. Az adatfeldolgozó komponens amely bemutatja az adat tisztítás folyamatát. Először a párhuzamosan futó GUI és Unicorn EEG szálaktól kapott adatok kerülnek külön listákba. Majd a külön válogatott adatokat ezután az MNE Epochs osztályok létrehozására alkalmas formátumúra alakítjuk. Végül a szűrések után megtörténik az epocholás és a modellek tanítása.

ban található Epochs osztályokat tudjuk velük létrehozni. Azért van erre szükség, mert így tudjuk kihasználni az MNE biztosította szűrési lehetőségeket. Először a két külön

szálon kapott szöveges formátumban tárolt időbélyegeket daraboljuk szét. Majd a `selected_index_timestamps` lista alapján szétszedjük az elemeket két külön listára. Az elsőben az intervallumok kezdeti időpontja, míg a másodikban a tanulás során használt aktuális kép index megjelenítésének végpontját tárolja. Ez azért kell, mert így tudjuk majd megmondani a modelleknek, hogy melyik is volt az a kép amit a felhasználónak néznie kellett. Az `eeg_data`, `data_timestamps` és a `trigger_timestamps` listákat megkapott intervallumok függvényében csoportosítjuk.

A felhasznált csatornák és mintavételezési megadásával frekvenciával létrehozunk egy úgynevezett Info objektumot amit az előállított adatokkal ötvözve `RawArray` típusú objektumokat készítünk. Mindezek után leszűrjük az agyhullámokat 1 és 20 Hz között, majd a jelből kiszűrjük az 50 Hz-es hálózati zajokat.

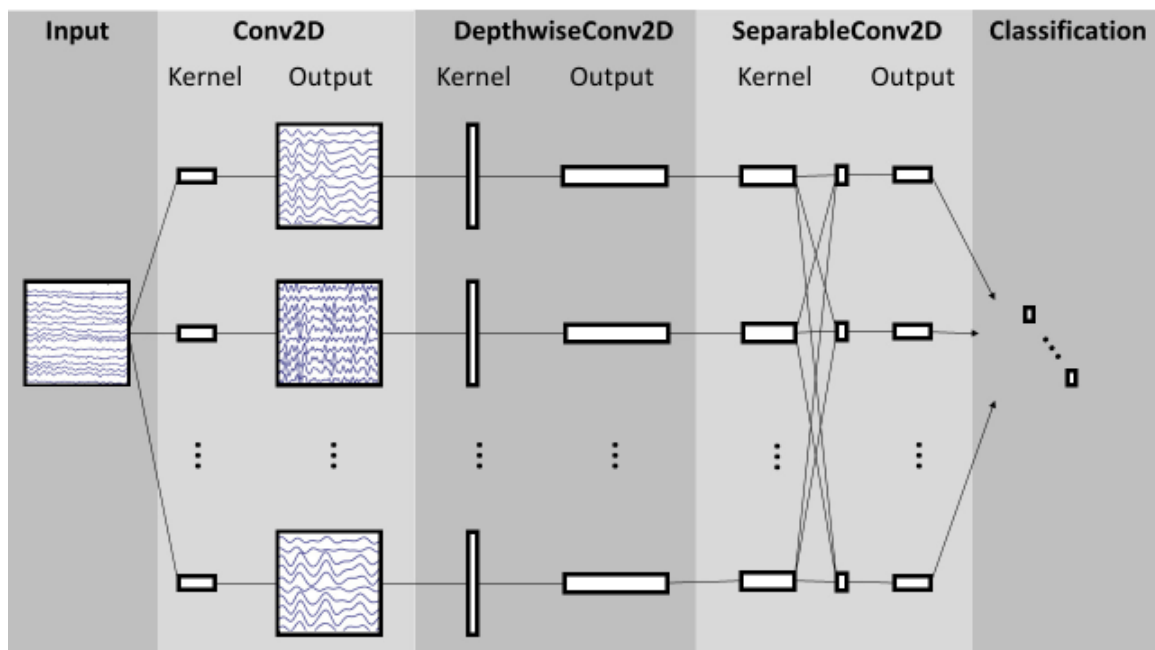
Az epocholást megelőző lépésként kigyűjtjük a villanásokból azokat az elemeket amikor azon képek villanak fel amiknek az indexe éppen a képernyőn volt megjelenítve. Ezekhez az elemekhez 1-es értéket, míg a többi intervallumhoz tartozó elemhez 0-t társítunk. Erre azért volt szükség, mert ezekkel a kigyűjtött adatokkal tudja majd megvizsgálni, hogy az adott epochban volt-e P300 jel. Összeálltak a szükséges elemek az epocholáshoz. Az újonnan létrehozott `Epochs` objektumokra alkalmazzuk a konfigurációs fájlban meghatározott „baseline”-t. Utolsó simításként kiszűrjük a hibás epochokat és egy dict típusú objektumba eltároljuk a letisztított és formázott adatokat.

A leszűrt és előkészített EEG adatok felhasználásával megkezdődik a modellek tanításának előkészítése. A kutatásom során két különböző típusú modellt használtam: az EEGNet-et valamint egy lineáris diszkriminanciaanalízis. Mindkét modell esetében a feldolgozott adatokból egy speciálisan erre célra készített és `MI.Raw` néven definiált, PyTorch-alapú Dataset osztályt használtam. Ez a dataset úgy lett megtervezve, hogy kompatibilis legyen a PyTorch-féle Dataloader osztállyal, lehetővé téve az adatok hatékony betöltését és kezelését. A Dataloader konfigurációjába beállítottam az adathalmaz méretére vonatkozó paramétereket, valamint egyedi mintavételezőt (sampler) alkalmazok, amely segít elkerülni a túltanulást, különösen a korlátozott adatmennyiség esetén.

6.9. Modellek tanítása

Az egyik választott modellem az EEGNet volt. Ez a modell a bemeneti agyi hullámokat különböző osztályokba sorolja, és számos paramétert, mint az osztályok számát, a csatornák és minták számát, a dropout rátát, a kernel hosszát és a szűrők számát tartalmazza.

Ahogy az az [6-15-es ábráról](#) is leolvasható, a modell három fő konvolúciós blokkból



6-15. ábra. Az EEGNet architektúrája, amely három fő konvolúciós blokkból áll. [49]

áll: Az első blokkban az időbeli konvolúciós réteg segítségével tanulja meg az egyes frekvenciaszűrőket, majd egy batch normalizációs réteg következik a tanulási folyamat stabilizálására. A mélységi konvolúciós blokkban minden jellemző térképet külön kezelnek, ahol mélységi konvolúciós rétegek segítségével tanulnak meg térbeli szűrőket, az ELU (Exponential Linear Unit) [50] aktivációs függvény használatával. Ezt követi egy átlagosító pooling és dropout réteg a túltanulás elkerülésére. A harmadik, szeparálható konvolúciós blokkban időbeli összefoglalást végzünk, majd egy pontonkénti konvolúcióval keverjük össze a jellemző térképeket, ami után egy másik átlagosító pooling és dropout réteg csökkenti tovább az adatok dimenzióit.

A modell az adatokat végül egy teljesen összekapcsolt rétegen keresztül osztályozza, és egy Softmax [51] réteggel számítja ki az egyes osztályokhoz tartozó valószínűségeket. A forward függvény meghatározza a modell előremenő útját, ahol a bemeneti adatok sorra áthaladnak a fent említett rétegeken, hogy kimeneti osztályozási valószínűségeket állítsanak elő.

A másik általam használt modell az LDA volt. A tanítás előtt az inicializálásakor kapott dict típusú adathalmazt a `concat_data` metódus dolgozza fel. Ez a függvény végigiterál az adathalmaz kulcsain, és a `MI_Raw` osztály segítségével betölti az egyes részhalmazokat, amelyeket aztán lapos formátumra alakít át, így létrehozva a független változókat tartalmazó `X` mátrixot és a célváltozókat tartalmazó `y` vektort. A tanítás után az LDA modell képes lesz meghatározni az osztályok közötti lineáris határvonalat.

7. Tesztelés

A szakdolgozatomban a program fejlesztése során különös hangsúlyt fektettem a kód működésének folyamatos tesztelésére, különösen az EEGNet és LDA modellek tanítási folyamataira. Az LDA modell esetében a széles körben elfogadott 80-20-as adatelosztást alkalmaztam, ahol az adatok 80%-át használtam a modell tanítására, míg a maradék 20%-ot tesztelési célokra fordítottam. Az EEGNet modellnél egy kiválasztott elemet használtam a teszteléshez a dictionary objektumból, míg a többi elemet a tanítási adathalmaz részeként kezeltem.

A modellek sikeres tanítása után a program éles változatának teszteléséhez fogtam hozzá. Mivel egy-egy kísérlet akár több tíz percet is igénybe vehetett, ezért a már előzetesen szűrt és tisztított kiexportált EEG adatokat tartalmazó pickle objektumokat olvastam be. Ez a módszer lehetővé tette számomra, hogy a kód bármely apró módosítása után ne kelljen újra és újra hosszadalmas tesztek végzését, így jelentősen csökkenthettem a fejlesztési ciklusok időigényét. Ezen kívül, a tesztelési folyamat során külön figyelmet fordítottam a modell általánosító képességének értékelésére is, hogy a valós körülmények közötti teljesítményét is megbizonyosodjam róla. Az adatokat több iterációban teszteltem, így tudtam elemezni a különböző konfigurációk és paraméterek hatását a modell pontosságára és hatékonyságára. Ennek köszönhetően a szakdolgozat részletesen dokumentálja a fejlesztési és tesztelési folyamat minden lépését, biztosítva ezzel a projekt átláthatóságát és reprodukálhatóságát.

8. Eredmények

Az EEGNet és az LDA modellek tanítása és éles tesztelése során rendkívül eltérő eredményeket tapasztaltam. Az EEGNet modell a tanulási fázisban kiemelkedően teljesített, átlagosan 90% feletti pontosságot érve el, ami a neurális hálók képességét bizonyítja, hogy milyen effektíven képesek megtanuli az adatokban fellelhető jellemzőket. Ezzel szemben az LDA modell, bár kevésbé volt pontos a tanulási fázisban, mégis jelentős, átlagosan 80% feletti pontosságot produkált, ami szintén tiszteletre méltó eredmény egy lineáris osztályozó esetében.

Ugyanakkor a valós környezetben történő tesztelés során az EEGNet teljesítménye jelentősen elmaradt az elvárásoktól. A modell megdöbbentő, csupán 10 és 20% közötti pontosságot tudott elérni. Ez arra utalhat, hogy a modell túlságosan rátanulhatott a tanítás során használt adatokra, ami a túlilleszkedés (overfitting) jelenségére utal. Ezzel szemben az LDA modell, bár kevésbé hatékony a tanulási fázisban, éles körülmények között 60 és 70% közötti pontosságot mutatott, ami jelzi az algoritmus robosztusabb természetét és jobb általánosító képességét.

Ezek az eredmények rávilágítanak arra a szükségességre, hogy további finomításokat és optimalizálást végezzek, különös tekintettel az EEGNet modell túlilleszkedésének csökkentésére. Ez magában foglalhatja az adathalmaz bővítését, a regularizációs technikák alkalmazását, vagy a hiperparaméterek további hangolását, hogy a modell jobban alkalmazkodjon az új, az EEGNet számára ismeretlen adatokhoz. Az LDA modell esetében pedig a további teljesítményjavítás érdekében érdemes lehet kiegészítő jellemzők kinyerését és szelekcióját fontolóra venni.

A szakdolgozatomban a továbbfejlesztési lehetőségeket illetően több újítást is fontolgatok, hogy a rendszer hatékonyságát és felhasználóbarátságát növeljem. Egyik ilyen terület a felhasználói felület továbbfejlesztése. Jelenlegi tapasztalataim azt mutatják, hogy egy hosszabb kísérlet során nehezen tudtam fenntartani a koncentrációm, ezért fontosnak tartom a felület ergonómiájának és interaktivitásának javítását.

Az SSVEP technológia bevezetését is tervezem, ami egy ígéretes módszer az agykutatásban. Az irodalomkutatás során világossá vált számomra, hogy az SSVEP alapú megközelítésekkel rövidebb kísérleti idő alatt lehetséges precízebb eredmények elérése. Bár a megvalósítás kihívásokkal teli lehet az eltérő frekvenciákon villogó képek szinkronizálása miatt, a potenciális előnyök, mint a gyorsabb válaszütem és a nagyobb pontosság, jelentősen javíthatják a rendszer teljesítményét.

Továbbá, az ERP megközelítést kiegészítő újításként úgy gondolom érdemes hangok-

hatásokkal társítani a vizuális ingerekhez. Az irodalom szerint ez javíthatja a pontosságot, mivel a multimodális ingerek feldolgozása általában erősebb agyi válaszokat vált ki. Ezt a megközelítést alkalmazva a célom egy olyan rendszer kifejlesztése, amely egy alap felhasználói szintű EEG sisakkal képes a lehető legpontosabb BCI létrehozására.

A projekt végső céljaként egy zenekészítő alkalmazáshoz szeretnék egy ilyen típusú interfészt implementálni. Mivel a zene különösen közel áll hozzám, fontosnak tartom, hogy lehetőséget biztosítsak azoknak az embereknek, akik fizikailag nem képesek ilyen programokat irányítani. Ezáltal nem csupán a technológiai innovációra törekszem, hanem a társadalmi inkluzivitás és az esélyegyenlőség előmozdítására is.

9. Összefoglaló

Az agy-számítógép interfész (BCI) egy olyan innovatív technológia, amely közvetlen kommunikációt tesz lehetővé az emberi agy és a számítógép között. Ez az úttörő megközelítés lehetőséget biztosít arra, hogy az emberek gondolataikkal irányítsák a számítógépes rendszereket. A non-invazív BCI rendszerek elterjedéséhez nagymértékben hozzájárult az elektroencefalográfia (EEG). Az EEG segítségével értékes betekintést nyerhetünk az agy működésébe, ami elősegíti a különböző neurológiai állapotok tanulmányozását, valamint részletes időbeli tájékoztatást nyújt az agyi aktivitás változásairól. Az EEG jelekben bizonyos ingerek hatására megjelenik egy P300 nevű hullám, amely kulcsfontosságú szerepet játszik a figyelem irányításában. Ezért a P300 hullám jelentős diagnosztikai és prognosztikai potenciállal bír az ember-számítógép interakcióban.

Szakdolgozatomban egy olyan grafikus BCI programot készítettem, amely agyhullámokkal vezérelhető. Az EEG jeleket egy g.tec Unicorn Hybrid Black (Unicorn) nevű eszközzel gyűjtöttem, amely alkalmas EEG adatok rögzítésére. A rendszerem négy kulcsfontosságú összetevőből áll: egy grafikus interfész, amely a kísérlet megjelenítéséért felelős; egy modul, amely az Unicorn eszközből gyűjt adatokat; egy adatfeldolgozó programrész, amely az előző két komponensből összegyűjtött adatokat elemzi; és végül a gépi tanulást megvalósító modellek.

Az alkalmazás fő célja az EEG jelek valós idejű feldolgozása és analízise, amelyhez az EEGNet nevű neurális hálózatot és a lineáris diszkriminanciaanalízist (LDA) használtam. Az EEGNet modell a tanulási fázisban 90% feletti pontosságot ért el, ami jól szemlélteti a neurális hálók EEG adatokat feldolgozó képességét. Ezzel szemben az LDA modell 80% feletti pontosságot mutatott a tanulási fázisban. Azonban valós körülmények között az EEGNet modell pontossága 10 és 20% közé csökkent, ami túlilleszkedésre utal, míg az LDA modell 60 és 70% közötti pontosságot ért el, ami nagyobb robusztusságát és jobb általánosító képességét bizonyítja.

A feltanított modelleknek köszönhetően az alkalmazás képes az EEG-vel vezérelve kiválasztani a képernyőn megjelenített hangszereket. A dolgozatomban bemutattam, hogy lehetséges egy BCI rendszer alacsony költségvetésű, otthoni környezetben történő megvalósítása. Ez a megközelítés jelentős előrelépést jelenthet, mivel az ilyen rendszerek hosszú távon hozzájárulhatnak a felhasználók életminőségének javításához.

10. Summary

The brain-computer interface (BCI) is an innovative technology that enables direct communication between the human brain and a computer. This groundbreaking approach allows people to control computer systems with their thoughts. The widespread adoption of non-invasive BCI systems has been greatly facilitated by electroencephalography (EEG). Through EEG, we can gain valuable insights into brain function, which aids in studying various neurological conditions and provides detailed temporal information on changes in brain activity. In EEG signals, a wave known as P300 appears in response to certain stimuli, playing a crucial role in attention regulation. Consequently, the P300 wave holds significant diagnostic and prognostic potential in human-computer interaction.

In my thesis, I developed a graphical BCI program that can be controlled by brain waves. The EEG signals were collected using a device called the g.tec Unicorn Hybrid Black (Unicorn), which is capable of recording EEG data. My system consists of four key components: a graphical interface responsible for displaying the experiment; a module that collects data from the Unicorn device; a data processing module that analyzes the data gathered by the first two components; and finally, machine learning models.

The primary goal of the application is the real-time processing and analysis of EEG signals, for which I used a neural network called EEGNet and linear discriminant analysis (LDA). During the training phase, the EEGNet model achieved an accuracy of over 90%, demonstrating the capability of neural networks to process EEG data. In contrast, the LDA model achieved an accuracy of over 80% during the training phase. However, in real-world conditions, the accuracy of the EEGNet model dropped to between 10% and 20%, indicating overfitting, while the LDA model achieved an accuracy of between 60% and 70%, proving its greater robustness and better generalization capability.

Thanks to the trained models, the application can select instruments displayed on the screen controlled by EEG signals. In my thesis, I demonstrated that it is possible to implement a BCI system in a low-budget, home environment. This approach represents a significant advancement, as such systems can contribute to improving the quality of life for users in the long term.

11. Hivatkozások

- [1] Z. Huang és M. Wang, „A review of electroencephalogram signal processing methods for brain-controlled robots”, en, *Cognitive Robotics*, 1. évf., 111–124. old., 2021, ISSN: 26672413. DOI: [10.1016/j.cogr.2021.07.001](https://doi.org/10.1016/j.cogr.2021.07.001). cím: <https://linkinghub.elsevier.com/retrieve/pii/S2667241321000094> (elérés dátuma 2023. 06. 03.).
- [2] R. Splittgerber, *Snell's Clinical Neuroanatomy*. Wolters Kluwer Health, 2018, ISBN: 978-1-975116-15-6. cím: <https://books.google.hu/books?id=7DRyDwAAQBAJ>.
- [3] R. S. Snell, *Clinical neuroanatomy*, en, 7th ed. Philadelphia: Wolters Kluwer Health/Lippincott Williams & Wilkins, 2010, OCLC: ocn253189573, ISBN: 978-0-7817-9427-5.
- [4] J. M. Frade és M. C. Ovejero-Benito, „Neuronal cell cycle: the neuron itself and its circumstances”, *Cell cycle*, 14. évf., 5. sz., 712–720. old., 2015.
- [5] K. A. Jellinger, „Niedermeyer's Electroencephalography: Basic Principles, Clinical Applications, and Related Fields, 6th edn”, *European Journal of Neurology*, 18. évf., 2011. cím: <https://api.semanticscholar.org/CorpusID:71335578>.
- [6] J. R. Wolpaw és E. W. Wolpaw, szerk., *Brain-computer interfaces: principles and practice*, en. Oxford ; New York: Oxford University Press, 2012, ISBN: 978-0-19-538885-5.
- [7] J. J. Shih, D. J. Krusienski és J. R. Wolpaw, „Brain-Computer Interfaces in Medicine”, *Mayo Clinic Proceedings*, 87. évf., 3. sz., 268–279. old., 2012. márc., ISSN: 0025-6196. DOI: [10.1016/j.mayocp.2011.12.008](https://doi.org/10.1016/j.mayocp.2011.12.008). cím: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3497935/> (elérés dátuma 2023. 05. 26.).
- [8] A. Kumari és D. R. Edla, „A Study on Brain–Computer Interface: Methods and Applications”, en, *SN Computer Science*, 4. évf., 2. sz., 98. old., 2022. dec., ISSN: 2661-8907. DOI: [10.1007/s42979-022-01515-0](https://doi.org/10.1007/s42979-022-01515-0). cím: <https://doi.org/10.1007/s42979-022-01515-0> (elérés dátuma 2024. 05. 14.).
- [9] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung és B. J. Lance, „EEGNet: A Compact Convolutional Network for EEG-based Brain-Computer Interfaces”, en, *Journal of Neural Engineering*, 15. évf., 5. sz., 56013. old., 2018. okt., arXiv:1611.08024 [cs, q-bio, stat], ISSN: 1741-2560, 1741-2552. DOI: [10.1088/1741-2552/aace8c](https://doi.org/10.1088/1741-2552/aace8c). cím: <http://arxiv.org/abs/1611.08024> (elérés dátuma 2023. 05. 20.).

- [10] S. Sakhavi, „APPLICATION OF DEEP LEARNING METHODS IN BRAIN-COMPUTER INTERFACE SYSTEMS”, dissz., 2017. jan. DOI: [10.13140/RG.2.2.32800.28166](https://doi.org/10.13140/RG.2.2.32800.28166).
- [11] H. Yu, S. Ba, Y. Guo, L. Guo és G. Xu, „Effects of Motor Imagery Tasks on Brain Functional Networks Based on EEG Mu/Beta Rhythm”, en, *Brain Sciences*, 12. évf., 2. sz., 194. old., 2022. jan., ISSN: 2076-3425. DOI: [10.3390/brainsci12020194](https://doi.org/10.3390/brainsci12020194). cím: <https://www.mdpi.com/2076-3425/12/2/194> (elérés dátuma 2023. 06. 03.).
- [12] O. A. Mokienko, L. A. Chernikova, A. A. Frolov és P. D. Bobrov, „Motor Imagery and Its Practical Application”, en, *Neuroscience and Behavioral Physiology*, 44. évf., 5. sz., 483–489. old., 2014. jún., ISSN: 0097-0549, 1573-899X. DOI: [10.1007/s11055-014-9937-y](https://doi.org/10.1007/s11055-014-9937-y). cím: <http://link.springer.com/10.1007/s11055-014-9937-y> (elérés dátuma 2023. 06. 03.).
- [13] Z. Qiu, B. Z. Allison, J. Jin és tsai., „Optimized Motor Imagery Paradigm Based on Imagining Chinese Characters Writing Movement”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25. évf., 7. sz., 1009–1017. old., 2017. júl., Conference Name: IEEE Transactions on Neural Systems and Rehabilitation Engineering, ISSN: 1558-0210. DOI: [10.1109/TNSRE.2017.2655542](https://doi.org/10.1109/TNSRE.2017.2655542).
- [14] J. Ding, G. Sperling és R. Srinivasan, „Attentional modulation of SSVEP power depends on the network tagged by the flicker frequency”, *Cerebral cortex (New York, N.Y. : 1991)*, 16. évf., 7. sz., 1016–1029. old., 2006. júl., ISSN: 1047-3211. DOI: [10.1093/cercor/bhj044](https://doi.org/10.1093/cercor/bhj044). cím: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1880883/> (elérés dátuma 2024. 05. 12.).
- [15] F. Beverina, G. Palmas, S. Silvoni, F. Piccione és S. Giove, „User adaptive BCIs: SSVEP and P300 based interfaces”, en,
- [16] S. J. Luck, *An introduction to the event-related potential technique*, en, Second edition. Cambridge, Massachusetts: The MIT Press, 2014, ISBN: 978-0-262-52585-5.
- [17] *P300 Waves for Single Subjects — Getting Started with EEG Data*. cím: https://www.cs.colostate.edu/eeeg/data/json/doc/tutorial/_build/html/p300_single_subject.html (elérés dátuma 2024. 05. 15.).
- [18] S. Sanei és J. A. Chambers, *EEG Signal Processing and Machine Learning*, en.
- [19] J. Polich, „Updating P300: An integrative theory of P3a and P3b”, *Clinical Neurophysiology*, 118. évf., 10. sz., 2128–2148. old., 2007, ISSN: 1388-2457. DOI: <https://doi.org/10.1016/j.clinph.2007.04.019>. cím: <https://www.sciencedirect.com/science/article/pii/S1388245707001897>.

- [20] G. H. Blowers, J. A. Spinks és D. T. L. Shek, „P300 and the anticipation of information within an orienting response paradigm”, *Acta Psychologica*, 61. évf., 2. sz., 91–103. old., 1986, ISSN: 0001-6918. DOI: [https://doi.org/10.1016/0001-6918\(86\)90024-7](https://doi.org/10.1016/0001-6918(86)90024-7). cím: <https://www.sciencedirect.com/science/article/pii/0001691886900247>.
- [21] *What is Machine Learning? — IBM*. cím: <https://www.ibm.com/topics/machine-learning> (elérés dátuma 2023. 05. 29.).
- [22] A. Antony, A. Bhattacharjee, S. Thakur, S. Bharadwaj, S. Sonawane és M. Tomar, „A review on efficient EEG pattern recognition using machine learning and deep learning methods and its application”, en, Mohali, India, 2022, 50012. old. DOI: <https://pubs.aip.org/aip/acp/article/2829425> (elérés dátuma 2023. 05. 13.).
- [23] P. P. Ippolito, *Feature Extraction Techniques*, en, 2019. okt. cím: <https://towardsdatascience.com/feature-extraction-techniques-d619b56e31be> (elérés dátuma 2023. 06. 04.).
- [24] *Documentation overview — MNE 1.7.0 documentation*. cím: <https://mne.tools/stable/documentation/index.html> (elérés dátuma 2024. 05. 12.).
- [25] *Introduction - Brainstorm*. cím: <https://neuroimage.usc.edu/brainstorm/Introduction> (elérés dátuma 2024. 05. 12.).
- [26] *User Documentation*, en. cím: <https://www.fieldtriptoolbox.org/documentation/> (elérés dátuma 2024. 05. 12.).
- [27] *OpenViBE Documentation: Use-case 1: EEG Signal analysis*. cím: http://openvibe.inria.fr/documentation/3.3.0/Doc_Mensia_AdvViz_UseCases_SignalAnalysis.html (elérés dátuma 2024. 05. 12.).
- [28] F. S. Bao, X. Liu és C. Zhang, „PyEEG: An Open Source Python Module for EEG/MEG Feature Extraction”, en, *Computational Intelligence and Neuroscience*, 2011. évf., 1–7. old., 2011, ISSN: 1687-5265, 1687-5273. DOI: [10.1155/2011/406391](https://doi.org/10.1155/2011/406391). cím: <http://www.hindawi.com/journals/cin/2011/406391/> (elérés dátuma 2024. 04. 24.).
- [29] *BrainVision Analyzer — Brain Products GmbH & Solutions*, en-US. cím: <https://www.brainproducts.com/solutions/analyzer/> (elérés dátuma 2024. 05. 13.).
- [30] F. Lotte, L. Bougrain, A. Cichocki és tsai., „A review of classification algorithms for EEG-based brain–computer interfaces: a 10 year update”, en, *Journal of Neural Engineering*, 15. évf., 3. sz., 31005. old., 2018. jún., ISSN: 1741-2560, 1741-2552. DOI: [10.1088/1741-2552/aab2f2](https://doi.org/10.1088/1741-2552/aab2f2). cím: <https://iopscience.org/>

- iop.org/article/10.1088/1741-2552/aab2f2 (elérés dátuma 2024. 04. 08.).
- [31] S. Balakrishnama és A. Ganapathiraju, „LINEAR DISCRIMINANT ANALYSIS - A BRIEF TUTORIAL”, en,
 - [32] R. Schirrmeister, J. Springenberg, L. Fiederer és tsai., „Deep learning with convolutional neural networks for EEG decoding and visualization: Convolutional Neural Networks in EEG Analysis”, *Human Brain Mapping*, 38. évf., 2017. aug. DOI: [10.1002/hbm.23730](https://doi.org/10.1002/hbm.23730).
 - [33] T. Wang, E. Dong, S. Du és C. Jia, „A Shallow Convolutional Neural Network for Classifying MI-EEG”, en, *2019 Chinese Automation Congress (CAC)*, Hangzhou, China: IEEE, 2019. nov., 5837–5841. old., ISBN: 978-1-72814-094-0. DOI: [10.1109/CAC48633.2019.8996981](https://doi.org/10.1109/CAC48633.2019.8996981). cím: <https://ieeexplore.ieee.org/document/8996981/> (elérés dátuma 2024. 05. 14.).
 - [34] J. Kawahara, C. J. Brown, S. P. Miller és tsai., „BrainNetCNN: Convolutional neural networks for brain networks; towards predicting neurodevelopment”, *NeuroImage*, 146. évf., 1038–1049. old., 2017. febr., ISSN: 1053-8119. DOI: [10.1016/j.neuroimage.2016.09.046](https://doi.org/10.1016/j.neuroimage.2016.09.046). cím: <https://www.sciencedirect.com/science/article/pii/S1053811916305237> (elérés dátuma 2024. 05. 14.).
 - [35] T. M. Ingolfsson, M. Hersche, X. Wang, N. Kobayashi, L. Cavigelli és L. Benini, *EEG-TCNet: An Accurate Temporal Convolutional Network for Embedded Motor-Imagery Brain-Machine Interfaces*, en, arXiv:2006.00622 [cs, eess, stat], 2020. máj. cím: <http://arxiv.org/abs/2006.00622> (elérés dátuma 2024. 05. 14.).
 - [36] *Unicorn Hybrid Black* — *g.tec medical engineering GmbH*. cím: <https://www.gtec.at/product/unicorn-hybrid-black/> (elérés dátuma 2024. 05. 14.).
 - [37] *Unicorn Hybrid Black* — *g.tec medical engineering GmbH*. cím: <https://www.gtec.at/product/unicorn-hybrid-black/> (elérés dátuma 2024. 04. 25.).
 - [38] *Graphical User Interfaces with Tk*, en. cím: <https://docs.python.org/3/library/tk.html> (elérés dátuma 2024. 05. 12.).
 - [39] *pygame* — *pygame v2.6.0 documentation*. cím: <https://www.pygame.org/docs/ref/pygame.html> (elérés dátuma 2024. 05. 12.).
 - [40] *Documentation* — *PsychoPy v2024.1.4*. cím: <https://psychopy.org/documentation.html> (elérés dátuma 2024. 05. 12.).
 - [41] *pyenv/pyenv*, original-date: 2012-08-31T06:57:52Z, 2024. máj. cím: <https://github.com/pyenv/pyenv> (elérés dátuma 2024. 05. 10.).

- [42] *Pipenv: Python Dev Workflow for Humans* — *pipenv 2023.11.16.dev0 documentation*. cím: <https://pipenv.pypa.io/en/latest/> (elérés dátuma 2024. 05. 10.).
- [43] R. C. Martin, *Agile software development, principles, patterns, and practices*, en, First edition, Pearson new international edition. Harlow: Pearson, 2014, ISBN: 978-1-292-02594-0.
- [44] *Python* — *time.time_ns() method*, en-US, Section: Python, 2019. szept. cím: https://www.geeksforgeeks.org/python-time-time_ns-method/ (elérés dátuma 2024. 05. 02.).
- [45] M. B. Pontifex és C. A. Coffman, „Validation of the g.tec Unicorn Hybrid Black wireless EEG system”, en, *Psychophysiology*, n/a évf., n/a. sz., e14320., eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/psyp.14320>, ISSN: 1469-8986. DOI: [10.1111/psyp.14320](https://onlinelibrary.wiley.com/doi/abs/10.1111/psyp.14320). cím: <https://onlinelibrary.wiley.com/doi/abs/10.1111/psyp.14320> (elérés dátuma 2023. 05. 15.).
- [46] G. S. Rodrigues, *Multithreading vs. Multiprocessing in Python*, en, 2020. szept. cím: <https://towardsdatascience.com/multithreading-vs-multiprocessing-in-python-3afeb73e105f> (elérés dátuma 2024. 05. 10.).
- [47] K. J. Wong, *Multithreading and Multiprocessing in 10 Minutes*, en, 2023. máj. cím: <https://towardsdatascience.com/multithreading-and-multiprocessing-in-10-minutes-20d9b3c6a867> (elérés dátuma 2024. 05. 02.).
- [48] *multiprocessing* — *Process-based parallelism*, en. cím: <https://docs.python.org/3/library/multiprocessing.html> (elérés dátuma 2024. 05. 10.).
- [49] S. Åsly, „Supervised learning for classification of EEG signals evoked by visual exposure to RGB colors”, dissz., 2019. jún. DOI: [10.13140/RG.2.2.13412.12165](https://doi.org/10.13140/RG.2.2.13412.12165).
- [50] D.-A. Clevert, T. Unterthiner és S. Hochreiter, *Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)*, en, arXiv:1511.07289 [cs], 2016. febr. cím: <http://arxiv.org/abs/1511.07289> (elérés dátuma 2024. 05. 12.).
- [51] H. Phillips, *A Simple Introduction to Softmax*, en, 2023. máj. cím: <https://medium.com/@hunter-j-phillips/a-simple-introduction-to-softmax-287712d69bac> (elérés dátuma 2024. 05. 12.).

Ábrajegyzék

3-1. Az idegesejt.....	4
3-2. EEG alapú BCI rendszer.....	6
3-3. EEG agyhullámok.....	9
3-4. A P300 jel	14
3-5. MNE Vizualizáció.....	17
3-6. A Unicorn fejhallgató	21
3-7. Az Unicorn fejhallgató elektródái	22
4-8. Szekvenciadiagram.....	25
5-9. Rendszerterv	28
6-10. Az alkalmazás grafikus kinézete.....	33
6-11. Párhuzamos programozás.....	35
6-12. Adatok gyűjtése	36
6-13. Adatok időbélyegzése	37
6-14. Adatfeldolgozás	38
6-15. EEGNet.....	40