

TDK-dolgozat

Berkics Péter

Moduláris szoftvercsomag idegsejtválaszok analízisére

Berkics Péter, 2. évfolyam

Pécsi Tudományegyetem

Természettudományi kar

Matematikai és Informatikai Intézet

Témavezető: Dr. Buzás Péter, egyetemi docens

Konzulens: Király Balázs , egyetemi tanársegéd

Kivonat

Jelen dolgozatban egy általam fejlesztett moduláris szoftvercsomagot mutatok be, amely jelenleg még fejlesztés, bővítés alatt áll. A szoftver biológiai vizsgálatok eredményeinek feldolgozásában, analizálásában és értelmezésében nyújt segítséget a PTE ÁOK Élettani Intézetében. A programcsomagot Matlab nyelven készítettem. A program rendelkezik grafikus felhasználói felülettel is, de emellett a Matlab parancssorából is hívható a grafikus felület kihagyásával, parancssori paraméterátadással.

A dolgozatban bemutatom a program grafikus felületét és a már elkészült modulokat. A grafikus felület tervezése és megvalósítása során a legfontosabb szempont a grafikus felület teljesen dinamikus generálása és a grafikus elemek futásidejű felhelyezése volt. Elvárás volt még a grafikus felülettel szemben, hogy új modul hozzáadásakor külön fájlban hozzáadható legyen a modul grafikus felülete.

A keret végrehajtó része szintén teljesen dinamikus, új modulok hozzáadásakor sem kell változtatni a kódon. A keret a grafikus felület megkerülésével is jól paraméterezhető.

Tartalomjegyzék

1. A kutatás célja, biológiai vonatkozásai, a felhasznált eszközök, módszerek és eljárások bemutatása	2
1.1. A kutatás összefoglalása, a célkitűzések ismertetése	2
1.2. A kutatás jelentősége	2
1.3. A kutatás során használt módszerek	3
1.3.1. Kísérleti állatok, műtéti előkészítés	3
1.3.2. Vizuális ingerek	3
1.3.3. Elektrofiziológia	4
1.3.4. Az adatok útja az idegsejtektől a felhasználóig	4
1.3.5. Post mortem szövettani vizsgálat	6
1.4. Az Expo program bemutatása	6
2. Az Expo XML adatstruktúra	7
2.1. Az Expo Matlab toolbox	7
3. A programmal szemben támasztott elvárások	8
3.1. Keretrendszer	8
3.2. A modulokkal kapcsolatos elvárások	8
3.2.1. Analog Loader	8
3.2.1.1. Az XML fileok feldolgozásáról	9
3.2.2. A triggerkereső	11
3.2.3. Spike Keresztkorreláció	11
4. A program dokumentációja	12
4.1. A programcsomag logikai felépítése	13
4.2. A keretrendszer	14
4.3. A program futása	15
4.4. AnalogLoader	18
4.5. SpikeCrossCorrelation	19
5. A kutatás folytatásának további lehetőségei	20

1. A kutatás célja, biológiai vonatkozásai, a felhasznált eszközök, módszerek és eljárások bemutatása

1.1. A kutatás összefoglalása, a célkitűzések ismertetése

Sokakat érdekel, hogy hogyan látják az állatok a színeket. A tudomány jelenlegi állása szerint az emberhez hasonló színlátással csak a főemlősök egy része rendelkezik. A többi emlősállat egy ennél egyszerűbb színlátással bír, amely a zöld-vörös színtévesztő emberekéhez hasonló. Az ilyen színlátás alapja a szemben található két különböző féle fényérzékelő receptor (ún. csapsejt), amelyek a spektrum kék, illetve zöld részére érzékenyek. A kutatók azt gondolják, hogy a főemlősök evolúciója során mutáció eredményeképpen jelent meg a harmadik csapsejt-típus, amely a vörös fényt érzékeléséért felelős. A csapok ingerléséből származó elektromos jeleket speciális ideghálózatoknak kell előkészíteni a színek észleléséért felelős agykéreg számára.

Kutatási programunkban azt vizsgáljuk, hogy hogyan épül fel és működik ennek az ideghálózatnak az ősi formája, amely az egyszerűbb színlátású emlősöknél, így például macskáknál található meg. Ehhez mélyen elaltatott kísérleti állatokon a szemből az agykéregbe vezető idegpályák különböző pontjain vizsgáljuk az idegsejtek elektromos válaszait a szemet érő színes ábrákra. Már sikerült olyan idegsejteket találnunk, amelyek "látják" a színeket, sőt fekete-fehér ábrákra nem is reagálnak. Feltételezhető, hogy mi, emberek is ezt az idegpályát "örököltük", így már azt is jobban értjük, honnan ered az emberi színlátás. Most azt igyekszünk kideríteni, hogy a már megismert idegsejtek melyik agyterületre küldik tovább a színről szóló információt.

A jelenlegi kutatásunk célja ezen sejtpopuláció receptív mezőit alaposabban jellemezni. A módszer lényege a vizuális ingerekkel kiváltott válaszok mérése. Mind a CGL-ben, mind a látókéregben egysejt-elektrofiziológiai vizsgálatokat végzünk a sejtek kromatikus és tér-időbeli válaszprofiljának mérésére.

1.2. A kutatás jelentősége

A macskák és a rágcsálók képesek színdiszkrimináció alapján tanult viselkedéses válaszokat adni, ami arra utal, hogy a színinformációt kortikális mechanizmusok hasznosítják. Kutatásaink segítenek eldönteni, hogy a színopponens jel (amely valószínűleg a retinában képződik) milyen feldolgozás után éri el a magasabb kérgi területeket. Ez egyrészt hozzásegíthet annak

megértéséhez, hogy az emlős agy miként hasznosítja ezeket.

Másrészt az ősi színlátó pálya tanulmányozása révén eredményeink hozzájárulnak a főemlős színlátás evolúciós alapjának megértéséhez. Az erre vonatkozó jelenlegi elméletek ugyanis nagyrészt arra a feltevésre építenek, hogy a szükséges ideghálózatok már a korai, dikromatikus emlősökben is léteztek, mielőtt a különálló "vörös" és "zöld" csapok megjelentek volna.

1.3. A kutatás során használt módszerek

A kutatás fő módszere a vizuális ingerekkel kiváltott, extracellulárisan elvezetett egysejt válaszok mérése altatott, immobilizált, lélegeztetett macskákon.

1.3.1. Kísérleti állatok, műtéti előkészítés

A kísérleteket a laboratóriumunk számára érvényben levő állatkísérletes engedély és a vonatkozó törvényi előírások betartásával végezzük. A kísérletekhez felnőtt macskákat ketamin (7 mg/kg) és xylazin (1 mg/kg) i.m. adásával altatjuk. Az állat légcsövét tracheakanül kerül a mesterséges lélegeztetés lehetővé tétele miatt, illetve a lágyéki artériába egy artériás kanült helyezünk el a vérnyomás folyamatos monitorozásáram valamint folyadék és drogok adására. Ezután paralízist váltunk ki, melyet folytonosan infundált vecuronium-bromiddal (0,15 mg/kg/h) érünk el. A mesterséges lélegeztetést O₂, N₂O és izoflurán keverékkel végezzük, amely egyben az anesztézia szintjét is fenntartja a kísérlet alatt. A vérnyomást, EKG-t és a kilégzett CO₂ - szintet állandóan monitorozzuk, a testhőmérsékletet termosztált fűtőlappal tartjuk állandóan.

1.3.2. Vizuális ingerek

A paralízis beállásakor a macska szeme ismeretlen szögben áll meg, a szem fókusztávolsága szintén nem ismert. Ezen adatokat a kísérlet kezdete előtt megmérjük, majd a szemeket kontaktlencsével fókusználjuk a számítógép monitorára. A vizuális ingereket ezen mutatjuk be. Az ingerek kontrasztját szinuszosan moduláljuk egy a receptív mezőre centrált kör alakú mezőben. Ezen kívül változtatni tudjuk az ingerek geometriáját, tér- és időbeli frekvenciáját és színét. A színeket úgy kalibráltuk, hogy azokkal a kék- és zöld-csapokat specifikusan ingerel-hessük (csapizoláló ingerek). Ezen kívül akromatikus (fekete-fehér) ingereket is használunk. A vizsgálathoz CRT monitort használunk, melynek felbontása 1024x768, 96 Hz-es képfrissítési frekvenciával.

1.3.3. Elektrofiziológia

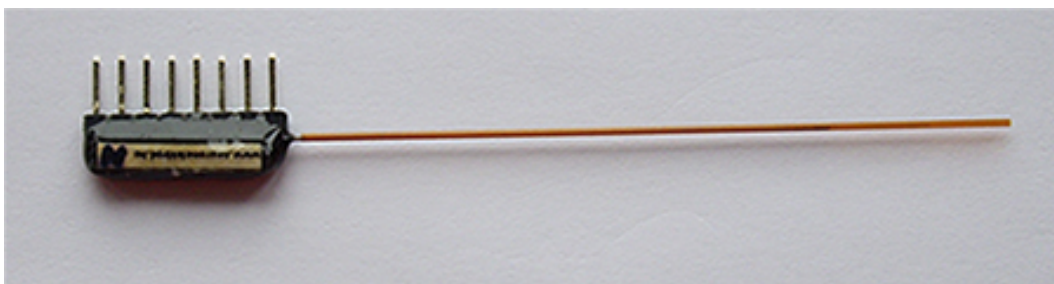
A koponyán a vizsgált terület fölött nyílást készítünk, majd mikromanipulátorok segítségével vertikálisan a célterületre juttatjuk az elektródát. Sztereotaxiás módszert alkalmazunk, melynek lényege hogy az agyról készült 1/10 mm pontosságú 3D atlasz segítségével tájékozódunk.

A neuronok akciós potenciál válaszait extracellulárisan mérjük. Ehhez egy a laborunk és egy hazai gyártó által közösen kifejlesztett, 7-csatornás mélyagyi mikroelektrodát használunk. A mikroelektroda egyes csatornái 15 μm átmérőjű platina-irídium drótból készültek, melyek egymástól kb. 20 μm távolságban helyezkednek el az elektróda hegyén. A többcsatornás elvezetés lehetővé teszi, hogy

- nagyobb biztonsággal különítsük el a különböző sejtek válaszait egymástól
- egyszerre több (rendszerint 4 vagy több) sejt válaszait regisztráljuk

A jeleket szűrjük (band-pass kb 500Hz - 4000Hz), erősítjük és digitalizáljuk. A potenciális akciós potenciálok hullámforma-mintázatait cluster analízis segítségével különítjük el és az izolációt kereszt-korrelogramok és a refrakter idő sértés alapján ellenőrizzük, ha kell módosítjuk. Az akciós potenciál sűrűségnek az inger frekvenciájával egyező harmonikus komponensét Fourier analízissel határozzuk meg. Ezt használjuk a sejt válasz mérőszámaként.

1.3.4. Az adatok útja az idegsejtektől a felhasználóig



1. ábra. Az elektróda (védőborításban)

A kísérlet során a fentebb leírt elektródát használjuk. (1. ábra). Az elektródából érkező jeleket a lehető legrövidebb vezetéken keresztül egy IC-erősítőbe vezetjük.

A rendszerben 8 csatornát használunk, az elektróda 7 csatornája mellett egy referenciaelektrodát is csatlakoztatunk az erősítőhöz, így a jeleket a referenciaelektrodához képest tudjuk rögzíteni. Ha a referenciaelektroda jeleit kivonjuk a többi csatorna jeleiből, akkor azzal ki tudjuk szűrni a különböző elektromos zajokat, amik amúgy befolyásolnák a vizsgálatot.

A jel ekkor még elég gyenge, mindössze néhány millivolt nagyságrendű, ezért továbbvezetjük a főerősítőbe, ahol nagyjából ezerszeresére erősítjük, így a kimeneten már Volt nagyságrendű jelet kapunk. Ezután egy frekvenciaszűrő következik, ahol egy analóg szűrőn bocsájtjuk át a jelet. Mivel mi az akciós potenciálokra vagyunk kíváncsiak, ezért a többi ingadozást megpróbáljuk elnyomni. Ennek eléréséhez sávszűrést alkalmazunk úgy, hogy az akciós potenciálok frekvenciatartományán szűrjük meg a jeleket (500-4000 Hz). Az így kondicionált jel: alapvo-
nal, azon zaj, amiből néha kiugranak tüskék.

A szűrt jelet ezután digitalizáljuk egy A/D-konverterrel (8 csatornás MOTU mikrofons erősítő, 2. ábra). A digitalizált jelet soros porton keresztül egy Apple Macintosh gépbe juttatjuk.



2. ábra. Az A/D-konverterként használt mikrofons erősítő

1.3.5. Post mortem szövettani vizsgálat

A kísérlet végeztével a kísérleti állatot túlaltatjuk, majd 4% paraformaldehiddel fixáljuk. A vizsgált agyterületből frontális metszeteket készítünk és krezil-ibolya festés után ellenőrizzük, hogy az elvezetés helye a célstruktúra mely részén volt.

1.4. Az Expo program bemutatása

A kísérlet folyamán a stimulust egy Apple Macintosh gép szolgáltatja, ezen fut az Expo program és erre a gépre érkeznek vissza az A/D-konverter által digitalizált jelek.

Az Expo program egy dr. Peter Lennie (New York University, USA) és más kutatók által kifejlesztett, nyílt forráskódú látásélettani kutatásokra kifejlesztett program. Az Expo 2 fő részre bontható:

- A kísérlet során használt vizuális ingerek generálása számítógép monitoron
- A kiváltott idegi válaszok regisztrálása

Az idegi válaszokat az Expo a számítógép audio bementén fogadja. Így a biológiai jelet bármilyen hangkártyával digitalizálhatjuk. Laborunkban erre jelenleg egy 8-csatornás mikrofon erősítő-konverter készüléket (MOTU 8pre, MOTU, USA) használunk. A digitalizált jelet a konverter FireWire kapcsolaton továbbítja a számítógépnek. A program lehetőséget ad a felvett válaszok általános célú elemzésére és megjelenítésére. A felhasználó speciális igényeit ez azonban gyakran nem elégíti ki. Összetettebb elemzési feladatokhoz az Expo adatstruktúrát XML formátumban exportálni lehet. Ennek szerkezetét a következő fejezetben mutatom be.

2. Az Expo XML adatstruktúra

Az Expo XML adatsuktúra igen összetett, a rögzített adatokon kívül tartalmazza a vizuális stimulusok definícióit és a kísérlet egyéb paramétereit. Ennek a struktúrának az a lényege, hogy visszakereshető belőle, hogy mely idegi válasz milyen szituációban lépett fel.

Az adatstuktúra kétféle adatot tartalmazhat, "Spike" és "Analóg" típusút:

- Spike típusú: A program által a kísérlet során már azonosított akciós potenciálok időbélyegei. Ebben az adattípusban minden akciós potenciált egy-egy időpont jellemez függetlenül annak eredeti jelalakjától. Ez a tárolási forma minimális memóriaigényű.
- Analóg típusú: Akkor használjuk ezt a tárolási formát, ha az idegi válasz jelalakját akarjuk vizsgálni. Az elnevezés félrevezető, hiszen ilyenkor nem az analóg adatfolyamot, hanem a digitalizált jelet rögzítjük folyamatosan. Ezen adatok kezelése nagy memóriaigényű.

2.1. Az Expo Matlab toolbox

Az Expo fejlesztői által rendelkezésünkre bocsájtott Matlab toolbox, amely tartalmazza az Expo XML adatok beolvasására és egyszerű keresésére szolgáló függvényeket. Bizonyos felhasználó-specifikus feladatokat meg lehet oldani ezek felhasználásával, másoknál viszont szükség volt az XML fileok közvetlen manipulációjára.

3. A programmal szemben támasztott elvárások

3.1. A keretrendszerrel kapcsolatos elvárások

A keretrendszerrel szemben 2 fő elvárás volt. Az egyik elvárás az volt, hogy a futtatási paraméterek megadásához a grafikus felületet biztosítsunk a felhasználó részére, a másik az volt, hogy parancssorból is futtatható legyen, a grafikus felület kihagyásával, a grafikus felület által bekért adatok parancssorból is könnyen átadhatóak legyenek. A parancssori futtatási lehetőség előrevetítí a későbbi köteget feldolgozási lehetőség megvalósítását is. Elvárás volt továbbá, hogy a program moduláris legyen, vagyis egy új modul hozzáadása, vagy egy régi modul továbbfejlesztése/cseréje ne befolyásolja a többi modul működését. A grafikus felülettel szemben az volt az elvárás, hogy a felhasználói felület teljesen dinamikusan jöjjön létre, csak a bemeneti fájlban végrehajtható analízismodulokat felajánlva. Elvárás volt még továbbá, hogy új fájlnev-marker vagy analízismodul esetén a felhasználói felület kódját ne kelljen módosítani, hanem egy külön fájlban hozzá lehessen adni az új modul grafikus felületét, illetve a már meglévő felület hiba nélkül működjön új markerekre is.

A prgoram megalkotása során ezen irányelveknek próbáltam megfelelni.

3.2. A modulokkal kapcsolatos elvárások

3.2.1. Az Analog Loader modullal szemben támasztott elvárások

Az analóg adat kibontó modul fejlesztésénél a bemeneti formátum egy XML file, amely base64-es kódolással tartalmazza az analóg adatot. Az expo környezethez készült Matlab csomagban van egy beépített függvény, ami ennek kibontására hivatott, azonban ez nagy adattömeg esetén memória hibát okozhat. A feladatunk egy olyan algoritmus kitalálása és Matlab környezetbe való implementálása volt, amely memóriahatékony és ugyanazt a kimenetet tudja generálni, mint a beépített függvények. Későbbi konzultációk során ezen követelményeket bővítettük, a bemenetet jobban paraméterezhetővé kellett tennem, a kimeneten pedig egyszerre több hasznos információt kellett elhelyeznem. A tervezés és megvalósítás közben ezen elvárásoknak próbáltam minél jobban megfelelni.

3.2.1.1. Az XML fileok feldolgozásáról. Az XML¹ formátum mára igen elterjedté vált, az adattárolás és üzenettovábbítás egyik standard fájlformátuma. Az XML fájlok szerkezetükben hierarchikusak, a tárolt elemek megfeleltethetők egy gyökeres fa csúcsainak. Az egymáshoz hasonló elemek (legyenek csúcsok vagy akár részfák) könnyen felismerhetők arról, hogy közös őssel rendelkeznek. Az XML fáklok lényegében egy soros fájl, amiben az értékes adatokat az adat típusát/minőségét jelző nyitó és záró tagek közé helyezzük, ez biztosítja a platformfüggetlenségét és könnyen értelmezhetőséget. Ezen előnyei miatt vált mára adattárolási és továbbítási szabvánnyá. Azonban pont a soros szerkezete, és az adattal egyszerre kódolt tagek jelentik a problémát. Az XML fájlok kezelése (különösen, ha nagy méretű XML fájlokról van szó) eléggé memóriaigényes.

A legtöbb XML-t használó alkalmazásban az XML fájlok elemzése és módosítása kulcsfontosságú. Ha a művelet adatmódosítást, véletlen elérést vagy nem a tárolási sorrendben való feldolgozást követel meg, akkor általában egy DOM²-on alapuló elemzőt használnak. Ennek a módszernek a hátránya, hogy a módszer által használt faszerkezet akár tízszer annyi helyet is foglalhat, mint az eredeti fájl, ami nehézkessé teheti vagy akár lehetetlenné is teheti a memóriában való tárolást, ezáltal a DOM technológia alkalmazását. Ezen probléma kiküszöbölésére alkalmas a SEDOM³. A módszer lényege, hogy az adatokat egy struktúrában tárolja. A struktúra segítségével megvalósítja, hogy az adatokat indexelt tárolókban tartsa a merevlemezen ezzel kiküszöbölve a memóriaproblémát, de továbbra is lehetőséget biztosít a gyors adatelérésre és módosításra.

Egy másik lehetőség a kulcsszavas keresés adatközpontú XML dokumentumokban a fa partícionálás felhasználásával. A módszer lényege, hogy felbontja a fát egymáshoz szervesen csatlakozó kompakt részfákra, amiket SI-fának neveznek, és ennek segítségével rögzíti az XML dokumentum strukturális információit. Az SI-fákat egy séma szabály alapján készítik. A jelentéstartalommal bíró SI-fák (MSI-fák) azonosításra kerülnek aszerint, hogy tartalmazzák-e a keresett kulcsszavak mindegyikét vagy legalább néhányat közülük. A visszakeresés gyorsítása érdekében az MSI fákat megindexelik a keresési kulcsszavak alapján. Az MSI fákat ezen kívül még a kulcsszavak és relevancia alapján rangsorolják is. A módszer hatékonyságát az is igazolja, hogy egy kulcsszavas keresés indítása 10-100 ms alatt eredményre vezet.

Más megoldások az XML fájlokat nyelvtani szempontból vizsgálja, felhasználva annak re-

¹Extensible Markup Language - Kiterjeszhető Jelölő Nyelv

²Document Object Modell

³Space Efficient XML DOM parser

guláris szerkezetét. Ezenkívül az XML fájlok ismert szerkezete miatt használható akár a soros beolvasás is, ennél azonban különösen kell figyelni az adatok és tagek megfelelő elkülönítésére, amit az adott alkalmazásban kell leprogramozni. Ha az alkalmazás csak az XML fájlok egy részét használja, akkor a számunkra értéktelen tagek és bennük elhelyezett adatok figyelmen kívül hagyhatóak. A soros feldolgozás implementálása bonyolult szerkezetű XML fájlok esetén nehézkes lehet.

Én a fent említett megoldások közül a soros olvasását választottam, egyrészt minimális memóriaigénye miatt, másrészt mert a fájlban található tagek közül csak néhány tartalmazott a modul számára értékes adatot.

3.2.2. A triggerkeresővel szemben támasztott elvárások

A triggerkeresővel kapcsolatban az volt az elvárás, hogy egy digitalizált analóg adatban tranziens csúcsokat (pl. akciós potenciálokat) keressen. Bemenetként az Analog Loader által beolvasott adatot használja, a kimenete pedig tranziensek időbélyegeit tartalmazó vektor legyen.

3.2.3. A spike keresztkorrelogram számító modullal kapcsolatos elvárások

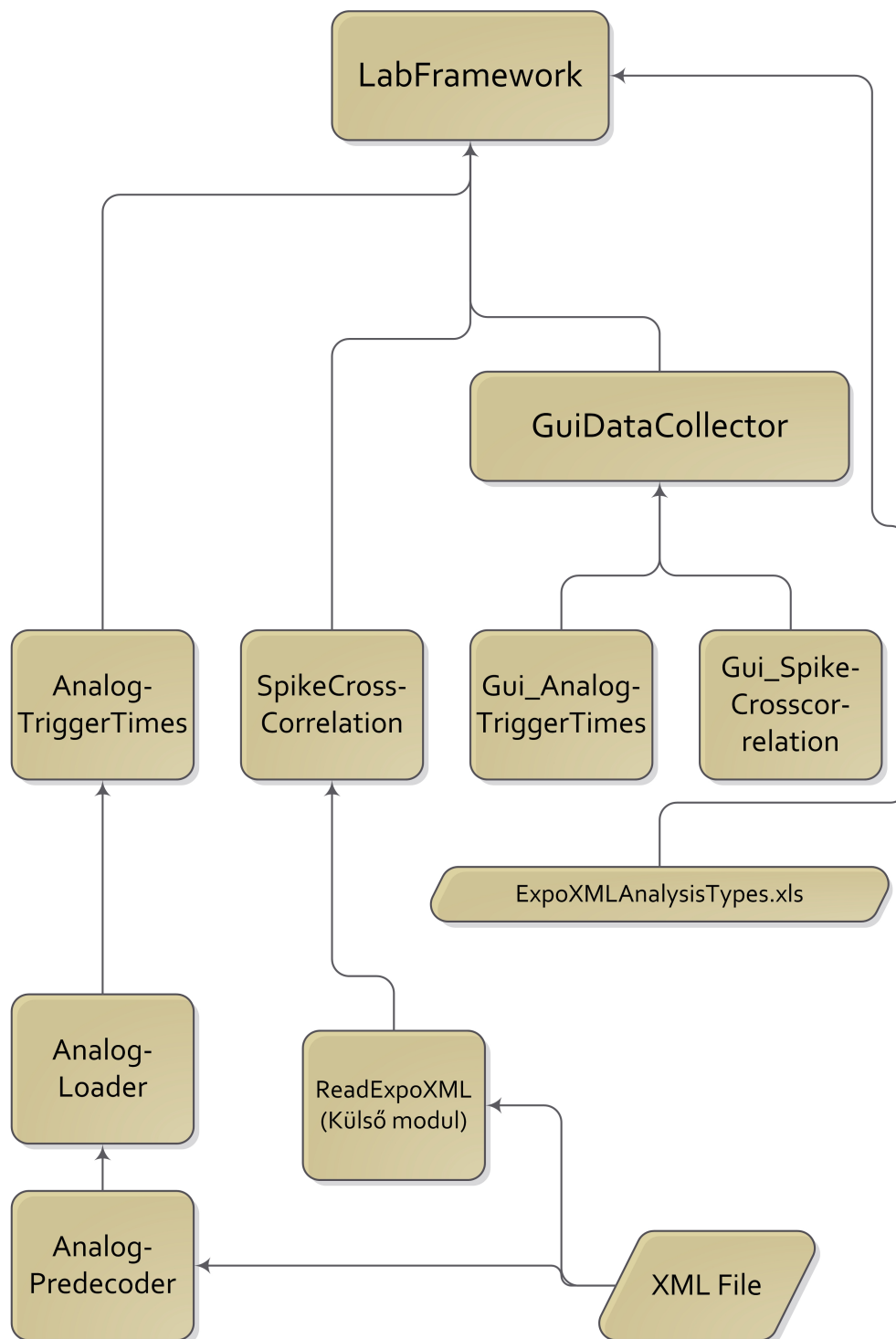
Az idegsejtválaszok elemzésénél alapkövetelmény, hogy a különböző idegsjetek párhuzamosan regisztrált és szuperponált válaszait megbízhatóan szétválasszuk. Ezt vagy az Expo programmal vagy egy külön programcsomaggal végezzük. A szétválasztás minőségét a kereszt- és autokorrelogramok vizsgálatával meg lehet ítélni.

A kereszt korrelogram megmutatja, hogy egy sejt akciós potenciáljai milyen valószínűséggel lépnek fel egy ettől különbözőnek ítélt sejt akciós potenciáljaihoz képest különböző időpontokban. Ha két sejt egymástól függetlenül tüzel, az egyik sejt tüzelési valószínűsége nem fog függni attól, hogy a másik sejt mennyi idővel előtte vagy utána tüzelt. Az autokorrelogram ugyanezt mutatja egyazon sejt akciós potenciáljain belül. Amennyiben az akciós potenciálok tényleg egy sejttől származnak, annak a valószínűsége, hogy az előző után újabb akciós potenciál jelenjen meg 2-3 ms-on belül nulla lesz az idegsejtmembrán refrakter periódusa miatt. Ez az autokorrelogramon egy 0 időpont körüli hiátus formájában jelenik meg.

4. A program dokumentációja

A dokumentációban bemutatom a keretrendszert és az elkészült modulokat.

4.1. A programcsomag logikai felépítése



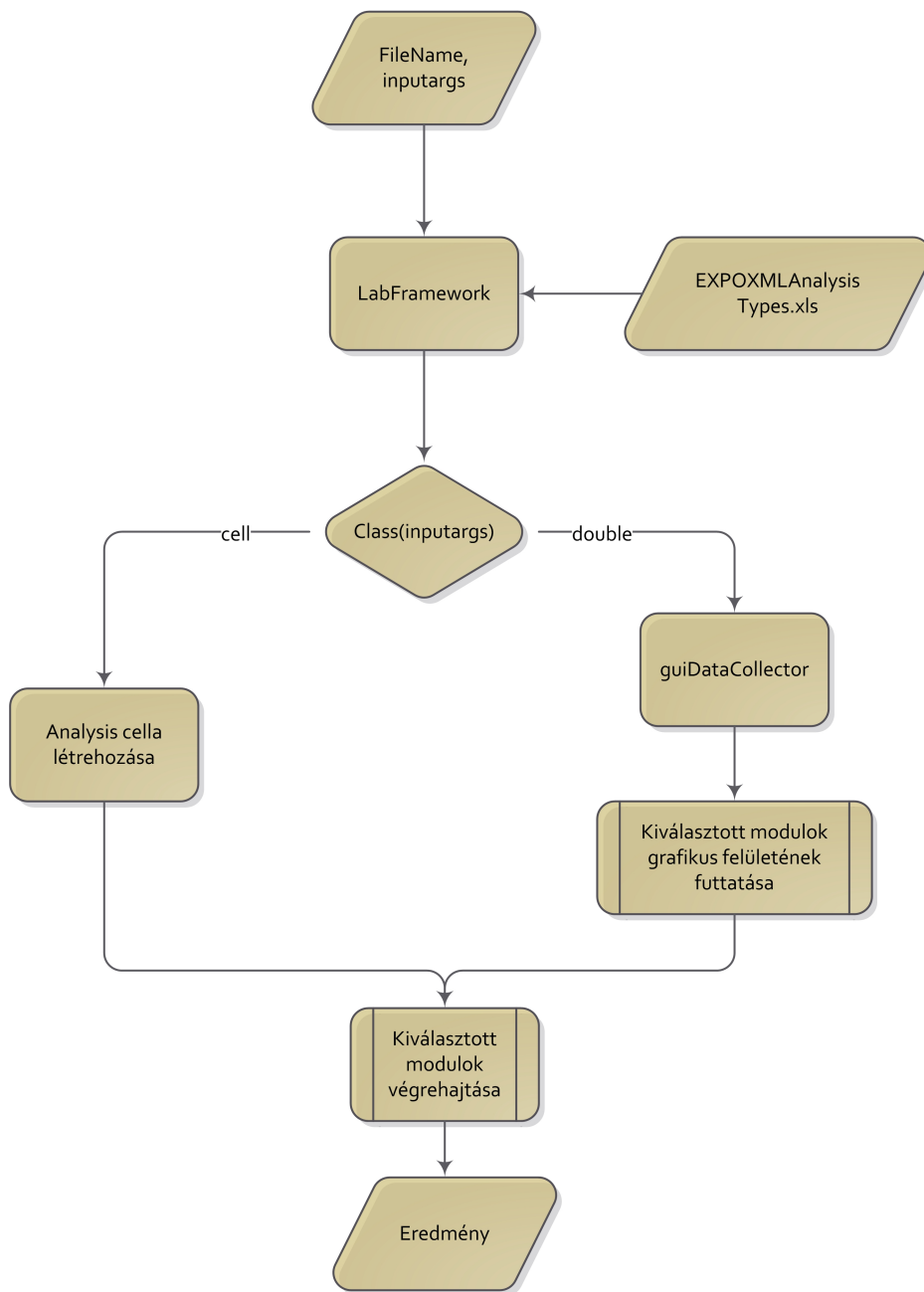
3. ábra. A modulok és külső fileok kapcsolata

4.2. A keretrendszer

```
Result=LabFramework (FileName,inputargs) ;
```

A LabFramework a programcsomag alapvető része. Bár a modulok futtathatóak önállóan is, de a LabFramework lehetőséget ad a felhasználónak, hogy a paraméterezést grafikusán oldja meg és egy fájlban több vizsgálatot futtasson le egymás után. A program felépítését a moduláris tervezés jellemzi, ez már a tervezés elején az egyik legfontosabb szempont volt. A modulok közötti kapcsolat az 3 ábrán látható.

4.3. A program futása



4. ábra. A program futása

A keretrendszer indításakor a keretnek meg kell adni egy fájlnevet és egy inputargs nevű második paramétert, amely kétféle lehet:

- vagy egy $2 \times n$ -es cellamátrix, melynek első sorában az analízismodulok nevei, második sorában a modulnak átadandó paraméterek vannak egy stuktúrában
- vagy egy 1-es (skaláris érték)

A második paraméter függvényében később szétválik a program végrehajtása.

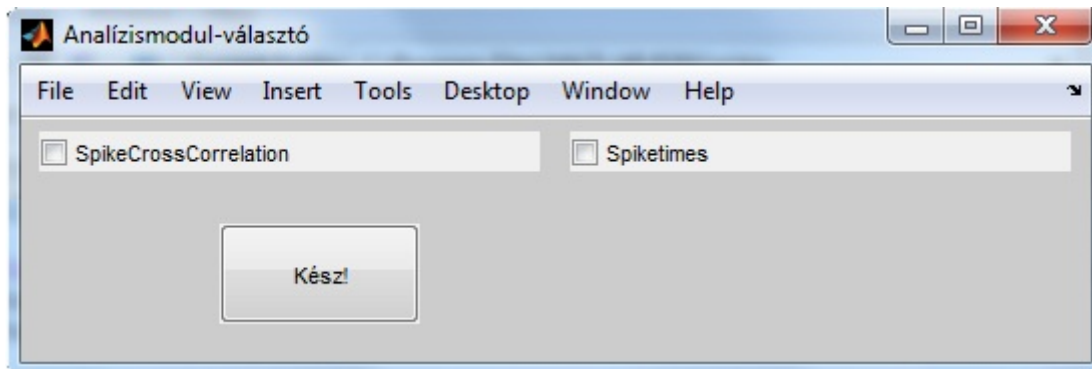
A fájlnevben szögletes zárójelek között a fájlban tárolt adatokra utaló markerek vannak, és ezen markerek alapján eldönthető, hogy az adott fájlban milyen analízismodulok végezhetőek el. Emiatt a LabFramework indításakor az első lépés a marker megkeresése a fájlnevben majd meghatározni, hogy az adott fájlra mely modulok hajthatók végre.

A könnyű bővíthetőség miatt a fájlnev markerek és a végrehajtható modulok a nyilvántartására egy külön fájlt használunk. Az adattárolásra excel táblázatot választottuk, hogy a felhasználóknak lehetősége legyen könnyen változásokat eszközölni ebben a fájlban. Erre azért van szükség mert bár az átlagfelhasználó nem fog új modult írni, viszont egy új fájl típuson valószínűleg a régi modulokat végre akarja hajtani.

A munkalapon egy olyan táblázat van, amelynek első sora (a második cellától kezdve) tartalmazza a modulok neveit, az első oszlopa pedig (a második cellától kezdve) tartalmazza a különböző filenév markereket. A táblázat többi része egy boolean mátrix, amelyben az egyesek jelentése, hogy a sorában található filenév markerre végrehajtható az oszlopában található modul.

	A	B	C	D
1		AnalogTriggerTimes	SpikeTimes	SpikeCrossCorrelation
2	CatCAQ_DKLContrast_sq	1	0	0
3	CatCAQ_DKLContrast_sq_noblank	1	0	0
4	CatCAQ_YposP	1	0	0
5	CatCAQ_YposN	1	0	0
6	CatCAQ_DirP	1	0	0
7	CatCAQ_DirN	1	0	0
8	CatCAQ_OriOrth_swapdir_sq	1	0	0
9	DU_get_Direction_cat	0	1	1
10	N_TempFreq_SvsML_cat	0	1	1
11	N_SpatFreq_SvsML_cat	0	1	1
12	N_Contrast_Ach_S_cat	0	1	1

5. ábra. ExpoXMLAnalysisTypes.xls



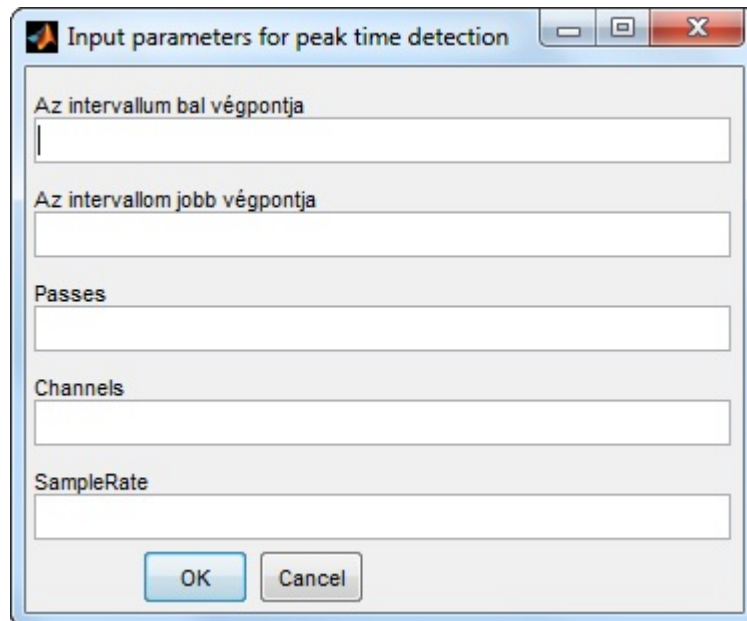
6. ábra. A modulválasztó

A keret futtatásakor először a táblázat feldolgozása történik, melynek során kikeressük, hogy a megadott fájlra milyen modulok végezhetőek. Ezután a feldolgozás kettéválik a grafikus felület futtatására vagy a parancssorban átadott paraméterek használatára.

A paraméteres feldolgozás esetén a megadott cellamátrix alapján létrehozuk az Analysis cellát. Ha a felhasználó által kért összes modul végrehajtható, akkor továbbmegyünk a megfelelő modulok futtatására (itt fog összeérni a grafikus és a parancsori mód), ha viszont nem hajtható végre, akkor a felhasználó számára jelződik a hiba és a program futása terminálódik.

A grafikus felülettel szemben támasztott elvárások teljesítéséhez azt a megoldást választottuk, hogy minden modul grafikus felülete külön fájlba kerül, azonos típusú bemeneti és kimeneti paraméterlistával, a keret grafikus része a modulválasztó ablakért lesz felelős illetve a modulválasztóban bejelölt moduloknak hívja meg a grafikus felületeit. Ezen résznél a függvények hívása dinamikusan, futásidőben lesz csak eldöntve. Emiatt az újonnan megírásra kerülő moduloknál egy szabályt alkalmazva kell elnevezni a modul grafikus felületét. A modul neve csak az angol abc betűit tartalmazhatja, nem tartalmazhat szóközöket, a szóhatárokat nagybetűk jelzik. A grafikus felület nevét úgy kapjuk, hogy a 'GUI_' stringet és a modul nevét összekonkatenáljuk. Például ha a modul neve AnalogTriggerTimes, akkor a grafikus felület neve kötelezően GUI_AnalogTriggerTimes kell, hogy legyen.

A grafikus módban először elindul a guiDataCollector. A guiDataCollector felületén bejelölheti a felhasználó, hogy mely modulok kerüljenek vérehajtásra (6. ábra), majd a modulok grafikus felületein megadhatja a modulok paramétereit. A grafikus mód itt tér át a modulok futtatására (itt fut össze újra a paraméteres futtatással)



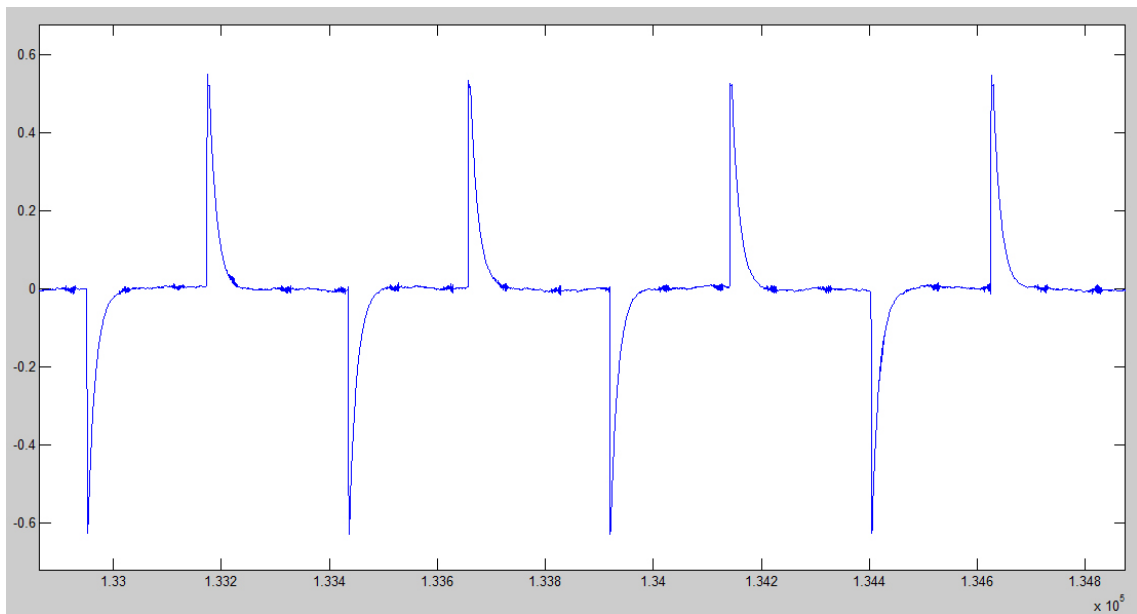
7. ábra. Példa egy modul grafikus felületére: AnalogTriggerTimes inputdialog

4.4. AnalogLoader

```
Result=AnalogLoader (D) ;
```

A D paraméter egy stuktúra, melynek a mezői a következők:

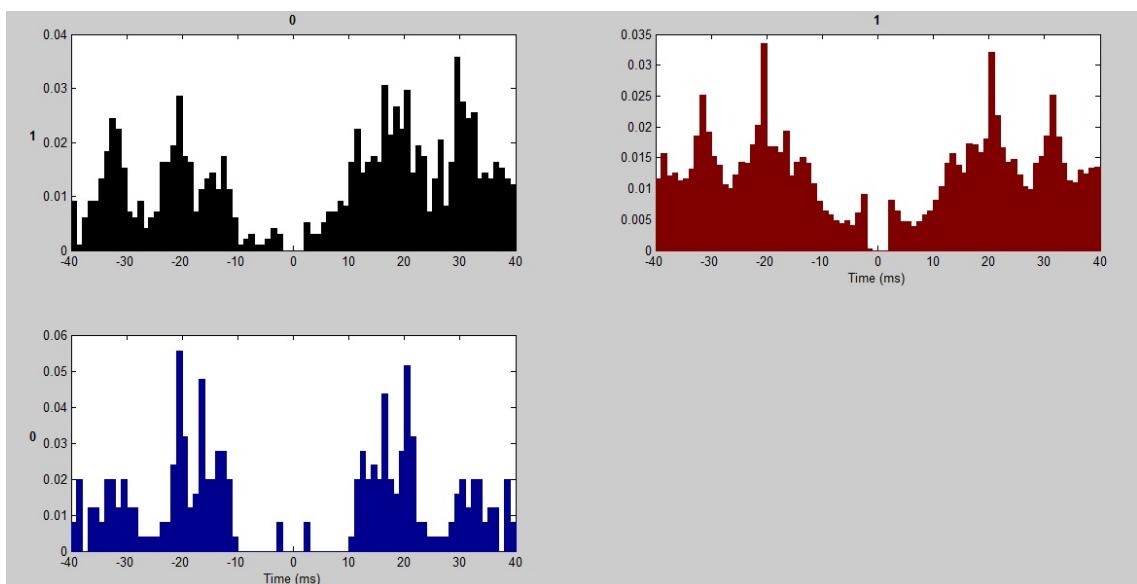
- FileName: Az analóg adatot tartalmazó Expo XML file neve
- Interval: A kibontani kívánt intervallum
- Pass: A kibontani kívánt Pass-ok azonosítója
- Subsample: Egy egész n érték, ha meg van határozva, akkor a kimenet csak minden n-edik mintát fogja tartalmazni
- Channel: A kibontani kívánt csatonák azonosítóját tartalmazó vektor



8. ábra. Az AnalogLoader outputjának grafikonja

4.5. SpikeCrossCorrelation

```
Result=f_SpikeCrossCorelation(D);
```



9. ábra. A keresztkorrelogramok

5. A kutatás folytatásának további lehetőségei

Eredményeink alapul szolgálnak az emlősök ősi színlátó pályájának teljes modelljének összeállításához. A vizsgálatokat két irányban lehet továbbvinni:

1. A retinális mechanizmusok vizsgálata. Ehhez pl. izolált retina sejtek fényválaszát kell mérni.
2. A kortikális mechanizmusok vizsgálata. Ehhez további agyterületeket kell megvizsgálni.

A vizsgálatok mindkét irányát célszerű lehet macskák helyett nappali életmódú rágcsálók (ürge, nyúl stb.) vagy mókuscickányon (*Tupaia*) végezni a macskákon kifejlesztett módszerekre és az ott kapott adatokra építve.

A kutatás során természetesen a programot is továbbfejlesztjük, új modulok kerülnek a már meglévők mellé.

Hivatkozások

- [1] Fangju Wang , Jing Li, Hooman Homayounfar
A space efficient XML DOM parser
Department of Computing and Information Science, University of Guelph, Guelph, Ont.,
Canada N1G 2W1
2006. January

- [2] LI Guoliang, FENG Jianhua, ZHOU Lizhu
Keyword Searches in Data-Centric XML Documents Using Tree Partitioning Department
of Computer Science and Technology, Tsinghua University, Beijing 100084, China
TSINGHUA SCIENCE AND TECHNOLOGY
ISSN11007-02141 102/ 211 lpp7-18
Volume 14, Number 1, February 2009

- [3] Stoyan Gisbert
Matlab - frissített kiadás
Typotex Kiadó, 2005
ISBN 9789639664821

- [4] Pascal Wallisch, Michael Lusignan, Marc Benayoun, Tanya I. Baker, Adam Seth Dickey,
Nicho Hatsopoulos
MATLAB for Neuroscientists: An Introduction to Scientific Computing in MATLAB
2009, Elsevier Inc
ISBN 978-0-12-374551-4

- [5] Brian R. Hunt, Ronald L. Lipsman, Jonathan Rosenberg
A guide to Matlab for beginners and experienced users
Cambridge University Press, 2006
ISBN 978-0-521-85068-1

- [6] Sergey E. Lyshevski
Engineering and Scientific Computations Using MATLAB
John Wiley & Sons, 2003
ISBN 978-0471462002
- [7] Duane C. Hanselman, Bruce L. Littlefield
Mastering MATLAB 7
2004
ISBN 978-0131430181
- [8] Buzás P, Kóbor P, Petykó Z, Telkes I, Martin PR, Lénárd L
Receptive field properties of color opponent neurons in the cat lateral geniculate nucleus
JOURNAL OF NEUROSCIENCE in press.