

به نام خدا

درس: سیگنال‌ها و سیستم‌ها

استاد: دکتر کربلایی

گزارش پروژه MATLAB شماره ۲

سید محمد امین منصوری طهرانی

۹۴۱۰۵۱۷۴

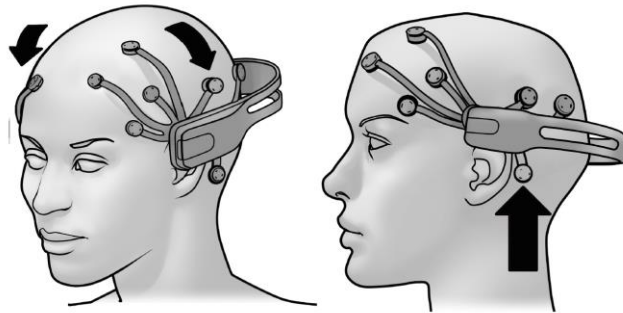
قسمت اول. توضیح داده و بررسی کلی آن

۱. توضیحاتی که در زیر می‌آید از دفترچه راهنمای دستگاه گرفته شده که در اینترنت موجود بوده است.
لینک دانلود:

https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&ved=0ahUKewj7mJLkI-SAhXrj5oKH54DC58QFgqeMAE&url=https%3A%2F%2Femotiv.zendesk.com%2Fhc%2Fen-us%2Farticle_attachments%2F200343895%2FEPOCUserManual2014.pdf&usq=AFQjCNENrIWCOgbQ-09IQqYXayBeeMF_g&sig2=zU4X6Ksk49Nbcmb3D6kXPQ&bvm=bv.150475504,d.bGs&cad=rja

Headset Placement

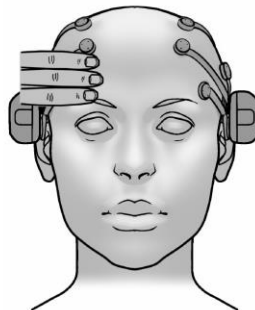
You are now ready to put the EPOC headset on your head. Using both hands, slide the headset down from the top of your head. Place the arms approximately as depicted, being careful to place the sensors with the black rubber insert on the bone just behind each ear lobe. Correct placement of the rubber sensor is critical for correct operation.



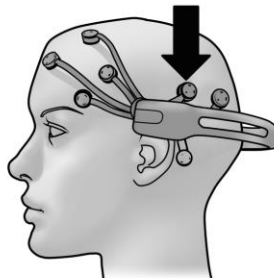
شکل ۱

EMOTIV

Notice the 2 front sensors should be approximately at the hairline or about the width of 3 fingers above your eyebrows.

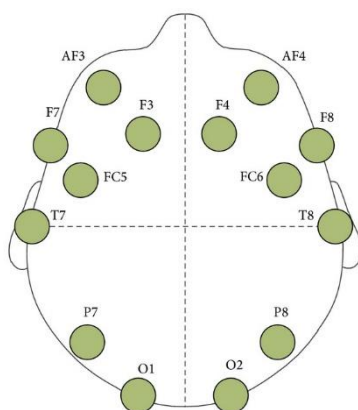


After the headset is in position, press and hold the 2 reference sensors (located just above and behind your ears) for about 5-10 seconds. **Good contact of reference sensors is the key for a good signal.** Check that the lights corresponding to these 2 reference sensors turn from red to green in the EPOC Control Panel Headset Setup screen.



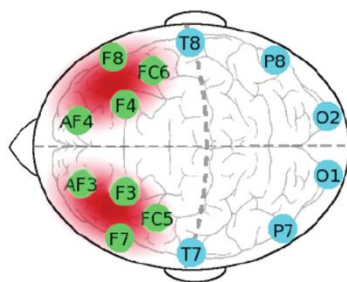
شکل ۲

مکان دقیق پروب ها روی سر در تصاویر ۱ و ۲ مشاهده می شود. در هر طرف سر ۷ پروب قرار دارد. مطابق دستور گفته شده در زیر هر تصویر، پروب هایی که لاستیک سیاه دارند، روی استخوان پشت گوش قرار می گیرند. سنسورهای جلویی تقریباً ۳ انگشت بالاتر از ابروها قرار می گیرند. بقیه سنسورها نیز روی شکل مشخص شده اند. پس از آن باید سنسورها را ابتدا فشرده و سپس به مدت ۵ تا ۱۰ ثانیه نگه داشته تا چراغ مربوطه سبز شود. دو تصویر زیر با جسجتو در اینترنت بدست آمده و لینک آنها در زیر قرار دارد.



شکل ۳

https://www.researchgate.net/figure/263394838_fig5_Diagram-showing-the-positions-of-the-14-channels-where-EEG-signal-was-recorded-by



شکل ۴

https://www.researchgate.net/figure/220506363_fig4_Figure-7-Sensor-placement-around-the-prefrontal-cortex-of-the-14-data-channels-in-the

۲. ابتدا برای دسترسی به فایل mat. با دستور `matfile(filename)` یک شی تعریف می کنیم. قبلاً با دستور `whos('-file','Train.mat')` اطلاعات فایل را می گیریم:

Name	Size	Bytes	class	Attributes
EEG	14979x16	1917312	Double	

جدول ۱

با دستور `who('-file','Train.mat')` نیز متغیرها را می بینیم:

Your variables are:

EEG

پس تنها متغیر ما همان *EEG* است. برای محاسبه موارد خواسته شده به داده‌های ستون‌های اول تا ۱۴ نیاز داریم. با استفاده از شی تعریف شده بردار *data* را می‌سازیم و با دستورات *mean, std, min, max* به ترتیب میانگین و انحراف معیار و مینیمم و ماکسیمم کانال‌ها را بدست می‌آوریم. نحوه عملکرد این توابع برای ماتریس‌ها به این شکل است که برداری سطری به عنوان خروجی بدست می‌دهند که هر ستون آن مقدار خروجی تابع است وقتی بر روی آن ستون در ماتریس ورودی اعمال می‌شود. نتایج در جدول ۲ مشاهده می‌شوند. ضمناً برای دقت بیشتر از دستور *format LONG* استفاده شده تا ۱۵ رقم نشان داده شود. (تمامی داده‌ها مانند داده‌های داده شده برحسب میکرو ولت هستند).

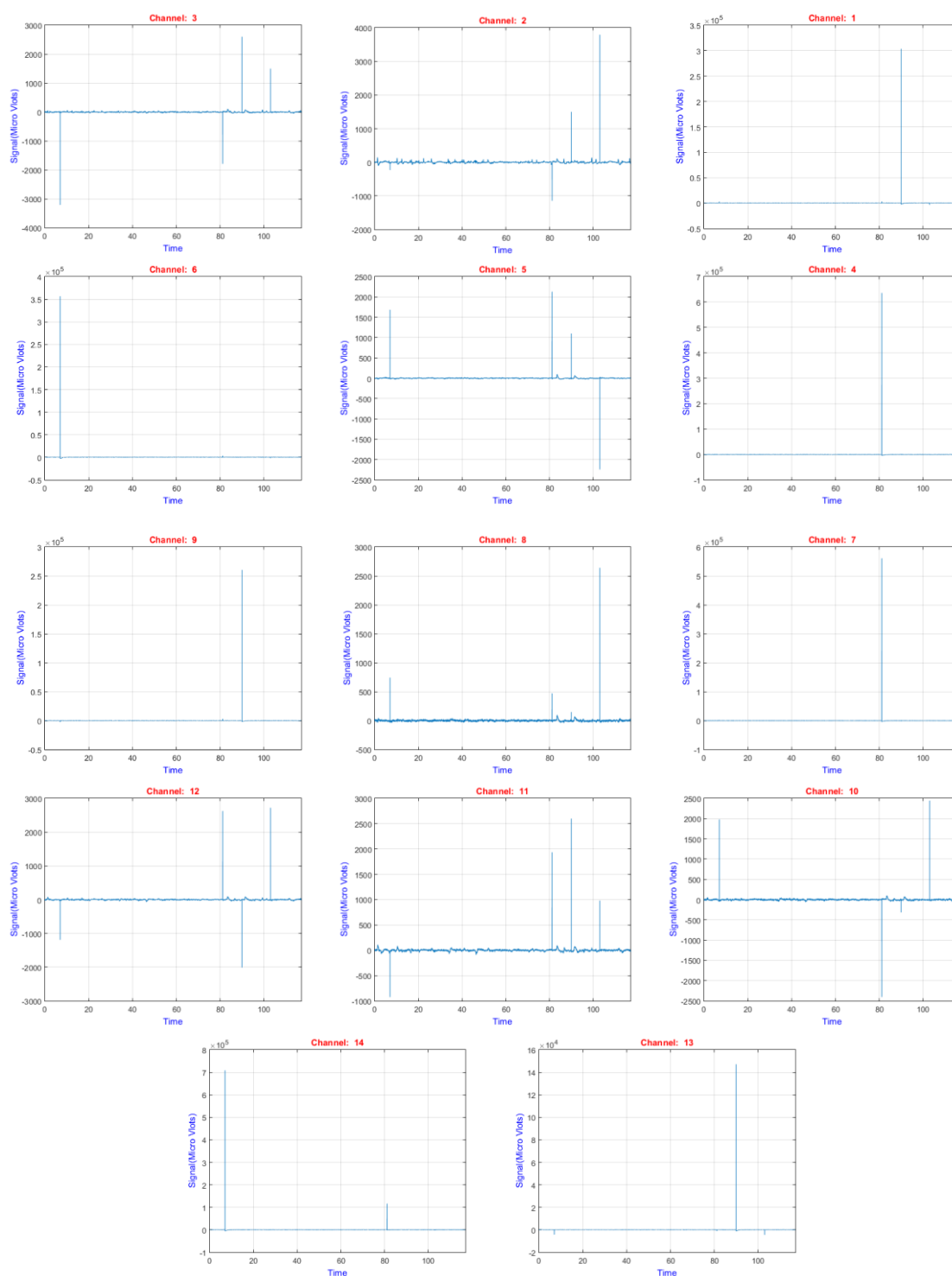
Channel	1	2	3	4
Mean	-0.128798738273142	0.197576975196768	-0.034013271331123	-0.293471211342027
Standard deviation	2487.215843354111	42.511876253510	41.807653157365	5206.490699180464
Min	-3239.082089341394	-1144.765985662501	-3211.267057832389	-4977.268184741650
Max	303738.9617271029	3792.4864423746	2602.0825815143	635954.3256629248
Channel	5	6	7	8
Mean	-0.066549135015740	-0.275587654848854	-0.360933701398763	0.009637458231431
Standard deviation	31.770584191398	2919.224950755706	4592.264589500382	25.879058468578
Min	-2242.391989422239	-2785.423774489010	-4388.645770378764	-38.619035238238
Max	2130.2365308074	356536.9925285098	560919.4468090813	2640.6323852604
Channel	9	10	11	12
Mean	-0.215618572481508	-0.116368299423987	-0.096998583870667	-0.047011339397907
Standard deviation	2132.310934911750	35.525990817028	33.890948754474	39.239041650457
Min	-2844.438770328796	-2403.354850525677	-923.765387199920	-2012.099016441398
Max	260416.9414143857	2445.4748976952	2600.6505960206	2722.3670283981
Channel	13	14		
Mean	-0.083968606865880	-0.459817534697260		
Standard deviation	1205.870486608183	5880.329182207291		
Min	-4486.584658195207	-5591.921131030559		
Max	147124.9713217138	708763.8829001824		

جدول ۲

با نگاه کلی به داده‌های فایل *mat*. واضح است مقادیر مینیمم و ماکسیمم داده‌های پرت هستند.

۳. با توجه به تعداد زیاد نمودارها، رسم آن‌ها با دستور *subplot* مناسب به نظر نمی‌رسد. (هر نمودار کوچک خواهد شد و اطلاعات زیادی بدست نخواهد داد). بنابراین با یک حلقه نمودارها را رسم می‌کنیم. ضمناً

برای نمایش مناسبتر، محدوده نمایش محور زمان را تا ۱۱۷ محدود کردیم. نتایج در تصاویر زیر مشاهده می‌شود. (به علت وجود عنوان در بالای هر نمودار در زیر تصویر شماره زده نشد).



۴. پس از جستجو، فایلی که در زیر لینک دانلود آن قرار دارد اطلاعات مناسبی را در اختیار گذاشت. در اینجا نکاتی کلی در رابطه با سیگنال *EEG* را که از این فایل بدست آمد به طور خلاصه می‌آوریم.

[https://www.brainmaster.com/software/pubs/brain/Benbadis%20EEG%20introduction%20\(Lee%20Chiong\)%20part%201.pdf](https://www.brainmaster.com/software/pubs/brain/Benbadis%20EEG%20introduction%20(Lee%20Chiong)%20part%201.pdf)

به طور کلی در کنار سیگنال *EEG* سیگنال‌های مصنوعی دیگری هم ضبط می‌شود که به ۲ دسته کلی *physiologic artifacts* و *extraphysiologic artifacts* تقسیم می‌شوند.

از دسته اول که مربوط به خود بدن هستند می‌توان به *EMG* که به ماهیچه‌ها مربوط است اشاره کرد. این سیگنال‌ها متداول‌ترین سیگنال از نوع اول هستند و فرکانس بالا محسوب می‌شوند. (۵۰ تا ۱۰۰ هرتز) از دیگر اثرات می‌توان به موارد زیر اشاره کرد:

- *Glossokinetic*: به حرکات دهان و مخصوصا زبان مربوط است.
- *ECG*: سیگنال‌های مربوط به قلب
- *Pulse*: وقتی سنسورها تصادفاً بر روی رگ قرار می‌گیرند. سیگنال‌های قلبی حدود ۲۰۰ تا ۳۰۰ میلی‌ثانیه زودتر از این سیگنال‌های پالس اتفاق می‌افتند و این سیگنال‌ها با سیگنال‌های قلب قابل تشخیص اند.
- *Eye Movement*: به حرکت کره چشم مربوطند و برای تشخیص مراحل خواب مفیدند.
- *Blink*: موردی که در این تمرین ما به دنبال آن هستیم. در اثر پلک زدن انحرافات مقارنی در سیگنال مغزی به وجود می‌آید.

برای دسته دوم می‌توان به گذرهای ناگهانی مقدار سیگنال آن‌هم تنها در یک یا دو کانال اشاره کرد که مشخصاً به فعالیت‌های مغزی مربوط نیستند.

در زیر به بررسی ۳ الگوی سیگنال مغزی می‌پردازیم.

◀ *Alpha Rhythm*: اولین چیزی است که خواننده سیگنال، آن را می‌بیند. فرکانس آن بین ۸ تا ۱۲ هرتز است. شکل کلی موج آن به صورت وزن دار و کم و زیادشونده است. دامنه آن بین ۲۰ تا ۱۰۰ میکرو ولت است. بهترین حالت برای مشاهده آن وقتی است که چشم بسته باشد. در صورتیکه چشم باز باشد این سیگنال ضعیف می‌شود.

◀ *Beta Activity*: فرکانس آن بین ۱۸ تا ۲۵ هرتز است. متقارن و وزن دار و کم و زیاد شونده است. دامنه آن بین ۵ تا ۲۰ میکرو ولت است. بیشتر در تشخیص مراحل اول و دوم خواب به کار می‌رود و در مراحل عمیق‌تر خواب ضعیف می‌شود.

◀ *Mu Rhythm*: فرکانس آن بین ۷ تا ۱۱ هرتز است و بیشتر شبیه حرف *m* است. معمولاً نامتقارن است و دامنه آن در محدوده نوع آلفا است.

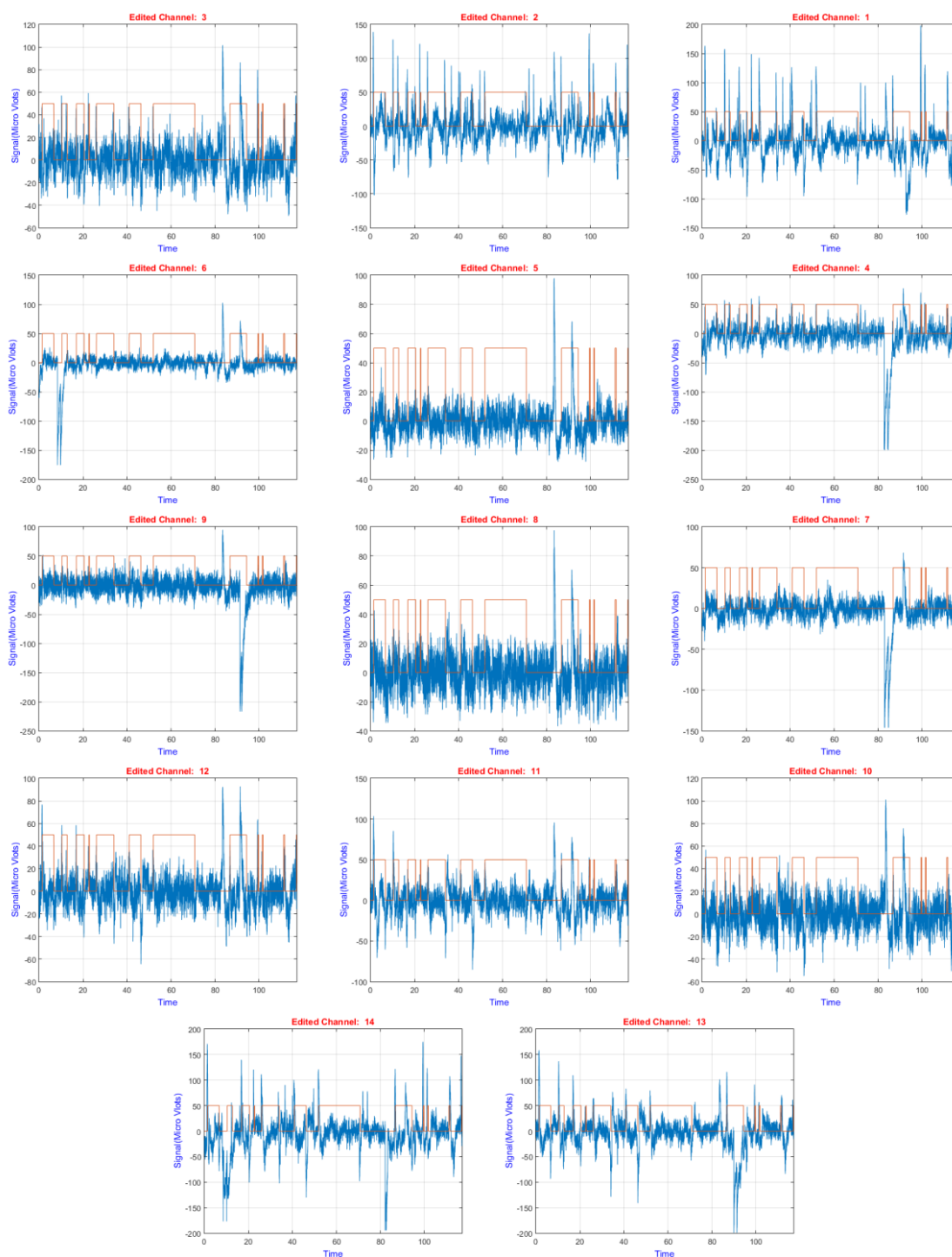
۵. مقادیر اکستریم بدست آمده با توجه به توضیحات بالا مشخصاً به فعالیت‌های مغزی مربوط نیستند. بنابراین باید با مقادیر معقولی جایگزین شوند. یکی از روش‌ها تقریب تکه ای خطی است. فرض می‌کنیم دو طرف داده

پرت تقریبا به صورت خطی به هم مربوط می‌شوند و لذا به جای داده پرت مقدار آن را متناظر با خط اخیر قرار می‌دهیم.

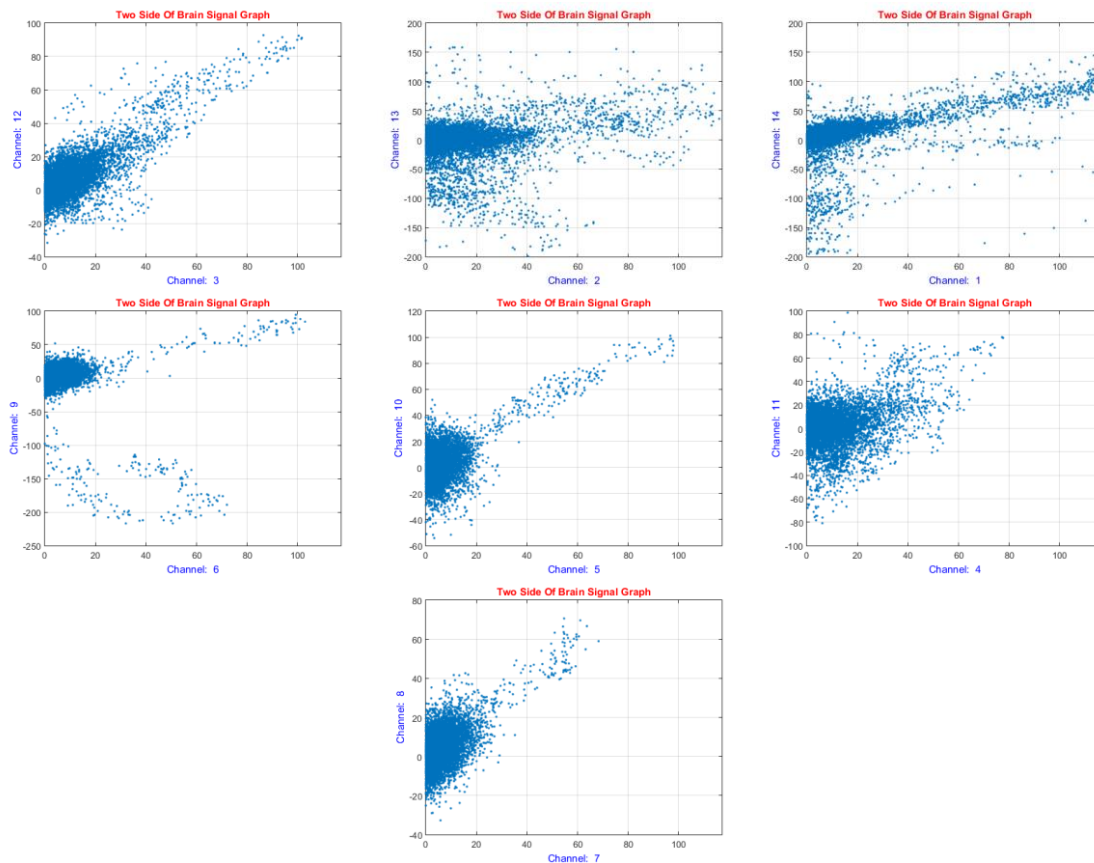
توضیح کد: ابتدا فایل اصلی داده ها و یک فایل جدید برای جایگزینی داده‌های پرت با داده‌های احتمالا صحیح‌تر لود می‌شوند. (ایجاد فایل جدید برای جلوگیری از رونویسی بر روی فایل اصلی است). سپس مقادیر اولیه داده‌های جدید را برابر داده‌های فایل اصلی قرار می‌دهیم. پس از آن در اولین مرحله از اصلاح داده‌های پرت، برای هر ۱۴ کانال به تعداد ۴ بار، هر بار مقادیر ماکسیمم و مینیمم کانال را می‌یابیم و سپس با تقریب قطعه ای خطی آن‌ها را با مقادیر صحیح‌تری جایگزین می‌کنیم. پس از این مرحله مشاهده می‌کنیم که در همه کانال ها داده‌های پرت تقریبا حذف شده‌اند. اما در کانال ۱۳ تعداد داده های پرت باقی مانده زیاد بوده و ۴ بار عملیات فوق کافی نمی‌باشد. برای اینکه زمان را تلف نکنیم، برای این کانال به طور جداگانه ۳۵ بار دیگر اصلاح فوق را انجام می‌دهیم. برای همه کانال‌ها از این‌جا به بعد برای حذف سایر داده‌های پرت چون استفاده از اعداد بیشتر در حلقه *for* زمان اجرای برنامه را به شدت افزایش می‌دهد، از مرحله دوم اصلاح استفاده کردیم.

مرحله دوم: در این مرحله برای کانال‌های باقی مانده مرتبا نقاط مینیمم و ماکسیمم یافت شدند و با مقادیر دیگری جایگزین شدند. علی‌رغم وقت‌گیر بودن، این کار به صورت دستی با جستجو و مشاهده داده‌ها در فایل Train.mat انجام گرفته است. اگر داده‌های پرت به صورت تجمع زیادی یافت شده است، با همان تعداد داده از دو طرف جایگزین شده است و داده‌های پرت تکی با داده کناری خود (چپ یا راست تفاوت چندانی ندارد) جایگزین شده‌اند. قبل از انجام این روش به این صورت عمل کردم که اگر مقادیر داده‌ها از بازه ۲۰۰- تا ۲۰۰ خارج می‌شد، آن‌ها را برابر صفر می‌گذاشتم. اما در قسمت‌های بعد به مشکل خوردم و لذا از روشی که گفته شد علی‌رغم زمان‌بر بودن استفاده کردم. جایگزینی با مقادیر نزدیک روش مناسبی است و احتمالا از بین رفتن داده در این روش کمتر است. ضمن اینکه این روش در صورتیکه فرض کنیم تغییرات سیگنال تا حد خوبی پیوسته‌اند و پرش‌های بسیار ناگهانی نداریم (سیگنال پلک زدن پرش‌های آنقدر ناگهانی هم تولید نمی‌کند). روشی معقول و منطقی می‌باشد.

۶. نتایج در تصاویر زیر مشهود است. شاخص باز یا بسته بودن در ۵۰ ضرب شده است.



۷. نتایج در تصاویر زیر قابل مشاهده است. بازهم به این دلیل که رسم در *subplot* امکان مشاهده جزئیات تصاویر را فراهم نمی‌کرد، از *plot* به صورت جداگانه استفاده شد. به نظر می‌رسد کانال‌های ۱ و ۱۴ و نیز بیشتر از آن‌ها کانال‌های ۳ و ۱۲ همبستگی خطی داشته باشند. بقیه کانال‌ها شاید همبسته باشند ولی همبستگی خطی در آن‌ها دیده نمی‌شود. ضمناً به جای رسم معمولی که نقاط به هم وصل شوند، برای خوانایی نمودارها، نقاط به هم وصل نشده‌اند.



۸. ماتریس ضرایب همبستگی در جدول ۳ نشان داده شده است. ماتریس مقادیر $p - value$ در جدول ۴ مشاهده می‌شود. اگر برای درایه ای از ۰/۰۵ کمتر باشد به معنی صحیح بودن همبستگی خطی است. بیشترین همبستگی مربوط به پروب‌های ۳ و ۱۲ می‌باشد. مقدار ضریب همبستگی برای آن‌ها ۰/۸۱ است. پس از آن‌ها پروب‌های ۱۱ و ۱۲ با ضریب ۰/۷۸۶ همبسته‌اند. سپس پروب‌های ۱۱ و ۱۰ با ضریب ۰/۷۴۹ همبسته‌اند. پروب‌های ۱۳ و ۱۱ نیز با ضریب ۰/۷۳۹ همبسته‌اند.

Columns 1 through 9

1.0000000000000000	0.607071309435467	0.688241935193006	0.430433658897313	0.194239192966767	-0.0107388801078879	-0.074781675728842	-0.012436460895199	0.247991520300765
0.607071309435467	1.0000000000000000	0.589500894229370	0.589652247512128	0.391971710414741	0.132524860248156	-0.123063351337680	-0.052364017576718	-0.082785052935634
0.688241935193006	0.589500894229370	1.0000000000000000	0.515814050574040	0.458075592026761	0.211340700886686	-0.001966015291687	0.283187536231276	0.08984825413914
0.430433658897313	0.589652247512128	0.515814050574040	1.0000000000000000	0.301213487730514	0.118801153844652	0.349007344938466	-0.066986726865963	-0.130666430638592
0.194239192966767	0.391971710414741	0.458075592026761	0.301213487730514	1.0000000000000000	0.614143887625043	0.124205348235868	0.473536784037455	0.1575543088995769
-0.0107388801078879	0.132524860248156	0.211340700886686	0.118801153844652	0.614143887625043	1.0000000000000000	0.211374488915515	0.451327746029130	0.171587319418177
-0.074781675728842	-0.123063351337680	-0.001966015291687	0.349007344938466	0.124205348235868	0.211374488915515	1.0000000000000000	0.257984814297952	0.020490206417095
-0.012436460895199	-0.052364017576718	0.283187536231276	-0.066986726865963	0.473536784037455	0.451327746029130	0.257984814297952	1.0000000000000000	0.431254410719928
0.247991520300765	-0.082785052935634	0.08984825413914	-0.130666430638592	0.1575543088995769	0.171587319418177	0.020490206417095	0.431254410719928	1.0000000000000000
0.286310593796215	0.075194804193422	0.455044022833764	0.025379872462203	0.514932942930041	0.377664807466273	0.167464890216327	0.704988567285896	0.445644593179509
0.559551559696173	0.177180327515234	0.564564391496841	0.137252886260821	0.39267601314843	0.190753084185870	0.079463974616048	0.445076619469861	0.322477180125058
0.661024773746570	0.383520821674862	0.810329680394843	0.319877564528671	0.383528299234649	0.433528299234649	0.243801707773820	0.076143329242321	0.191884605593495
0.698412382792063	0.182463150594259	0.475177209719342	0.088028085899211	0.158310313484164	0.000597985039458	-0.086978485561224	0.183636007464783	0.482968634859441
0.660813036105438	0.350059539054371	0.505898239793315	0.395071185917936	0.193499134661786	0.271875673529870	0.179898253849169	0.052409892122701	0.052953409249799

Columns 10 through 14

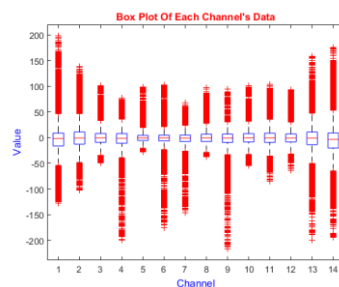
0.286310593796215	0.559551559696174	0.661024773746570	0.698412382792063	0.660813036105438
0.075194804193422	0.177180327515234	0.383520821674862	0.182463150594259	0.350059539054371
0.455044022833764	0.564564391496841	0.810329680394843	0.475177209719342	0.505898239793315
0.025379872462203	0.137252886260821	0.319877564528671	0.088028085899211	0.395071185917936
0.514932942930041	0.39267601314843	0.433528299234649	0.158310313484164	0.193499134661786
0.377664807466273	0.190753084185870	0.243801707773820	0.000597985039458	0.271875673529870
0.167464890216327	0.079463974616048	0.076143329242321	-0.086978485561224	0.179898253849169
0.704988567285896	0.445076619469861	0.4458421652402192	0.153636007464783	0.052409892122701
0.445644593179509	0.322477180125058	0.191884605593495	0.482968634859441	0.052953409249799
1.0000000000000000	0.749965749627353	0.677598067835122	0.566057704145386	0.354849932668420
0.749965749627353	1.0000000000000000	0.786126124120686	0.739056744263125	0.516009440171488
0.677598067835122	0.786126124120686	1.0000000000000000	0.626461298413281	0.593764883144982
0.566057704145386	0.739056744263125	0.626461298413281	1.0000000000000000	0.541929348673367
0.354849932668420	0.516009440171488	0.593764883144982	0.541929348673367	1.0000000000000000

1.0000000000000000	0	0	0	0.0000000000000000	0.188766669363358	0.0000000000000000	0.128005466813983
0	1.0000000000000000	0	0	0.0000000000000000	0	0.0000000000000000	0.000000000143030
0	0	1.0000000000000000	0	0	0.0000000000000000	0.809866501275879	0.0000000000000000
0	0	0	1.0000000000000000	0.0000000000000000	0.0000000000000000	0	0.0000000000000000
0.0000000000000000	0	0	0.0000000000000000	1.0000000000000000	0	0.0000000000000000	0
0.188766669363358	0.0000000000000000	0.0000000000000000	0.0000000000000000	0	1.0000000000000000	0.0000000000000000	0
0.0000000000000000	0.0000000000000000	0.809866501275879	0	0.0000000000000000	0.0000000000000000	1.0000000000000000	0.0000000000000000
0.128005466813983	0.000000000143030	0.0000000000000000	0.0000000000000000	0	0	0.0000000000000000	1.0000000000000000
0.0000000000000000	0.0000000000000000	0.0000000000000000	0.0000000000000000	0.0000000000000000	0.0000000000000000	0.012147780212275	0
0.0000000000000000	0.0000000000000000	0	0.001894415770234	0	0	0.0000000000000000	0
0	0.0000000000000000	0	0.0000000000000000	0	0.0000000000000000	0.0000000000000000	0
0	0	0	0	0	0.0000000000000000	0.0000000000000000	0
0	0.0000000000000000	0	0.0000000000000000	0.0000000000000000	0.941662463306720	0.0000000000000000	0.0000000000000000
0	0	0	0	0.0000000000000000	0.0000000000000000	0.0000000000000000	0.00000000013784

0.0000000000000000	0.0000000000000000	0	0	0	0
0.0000000000000000	0.0000000000000000	0.0000000000000000	0.0000000000000000	0	0
0.0000000000000000	0	0	0	0	0
0.0000000000000000	0.00189415770234	0.0000000000000000	0.0000000000000000	0	0
0.0000000000000000	0	0	0	0.0000000000000000	0.0000000000000000
0.0000000000000000	0	0.0000000000000000	0.0000000000000000	0.941662463306720	0.0000000000000000
0.012147708212275	0.0000000000000000	0.0000000000000000	0.0000000000000000	0.0000000000000000	0.0000000000000000
0	0	0	0	0.0000000000000000	0.0000000000137845
1.0000000000000000	0	0	0.0000000000000000	0	0.0000000000088785
0	1.0000000000000000	0	0	0	0
0	0	1.0000000000000000	0	0	0
0.0000000000000000	0	0	1.0000000000000000	1.0000000000000000	0
0	0	0	0	1.0000000000000000	0
0.0000000000088785	0	0	0	0	1.0000000000000000

جدول ۴

۹. *boxplot* همانطور که در صورت سوال گفته شده امکان مشاهده همزمان توزیع فراوانی چند دسته داده را فراهم می‌کند. اگر یک ماتریس به عنوان ورودی به این تابع بدهیم، در خروجی به ازای هر ستون این ماتریس یک نمودار جعبه ای خواهیم داشت که توزیع فراوانی داده های آن ستون را نشان می‌دهد. در تصویر ۵ نمودار جعبه ای سیگنال هر کانال پس از حذف شدن داده‌های پرت نشان داده شده است. خط قرمز میانی نشان دهنده میانه است. ضلع پایین مستطیل، چارک اول و ضلع بالایی چارک سوم را نشان می‌دهد. دو خط بعدی در انتهای خط‌چین‌ها مجاور بالا و مجاور پایین را نشان می‌دهند. (*upper & lower adjacents*). طبق تعریف مجاور بالا بزرگترین داده‌ای است که کوچکتر یا مساوی *UIF* است. *UIF* برابر چارک سوم به‌علاوه ۱/۵ برابر اختلاف چارک سوم و اول یا همان طول مستطیل می‌باشد. تعریف مشابهی برای مجاور پایین برقرار است.



شکل ۵

قسمت دوم. ویژگی پلک زدن ها

۱. ابتدا از روی یکی از تصاویر برحسب زمان که در قسمت‌های قبل رسم کرده‌ایم، تعداد تغییر حالت‌ها را می‌شماریم. ۱۱ باز و بسته شدن داریم که به معنی ۲۲ بار باز یا بسته شدن پلک می‌باشد. در آخر نیز یک بسته شدن پلک داریم که باز شدن آن در خارج از داده‌ها است. پس آرایه ای ۲۳ عضوی برای ذخیره زمان‌ها

تشکیل می‌دهیم و مقدار اولیه درایه‌های آن را صفر می‌گذاریم. سپس شمارنده درایه این آرایه را ایجاد می‌کنیم و مقدارش را یک می‌گذاریم. در حلقه‌ای که بعد از دستورات فوق قرار می‌گیرد، در ستون ۱۵ ماتریس اصلی به دنبال تغییر حالت‌ها می‌گردیم و این زمان‌ها را در آرایه گفته شده ذخیره می‌کنیم. نتایج در جدول ۵ مشاهده می‌شوند.

1.4531250000000	6.7890625000000	10.4218750000000	12.7812500000000	16.9843750000000
20.5546875000000	22.6406250000000	22.8515625000000	26.0937500000000	33.9843750000000
40.9531250000000	46.2968750000000	51.9609375000000	70.7187500000000	86.7421875000000
94.3281250000000	99.4218750000000	99.7578125000000	101.3593750000000	101.7656250000000
111.0546875000000	111.6171875000000	116.8515625000000		

جدول ۵

۲. در این قسمت برای مشاهده تغییرات برای هر کانال حول لحظات پلک زدن داده‌ها را بررسی کردیم. ابتدا برای ۴ کانال اول حول این لحظات جابجا شدم تا تغییرات را ببینم. در لحظات بسته شدن در بازه‌ای قبل از بسته شدن، سیگنال مقادیر زیادی می‌گرفت و پس از آن مقادیر منفی نسبتاً زیادی (اما کمتر از مقادیر مثبت قبل از بسته شدن) می‌گرفت. برای هر بار بسته شدن و برای ۴ کانال اول این بازه‌ها را بدست آوردم. طول بازه‌ها نیز به صورت حدودی در نظر گرفته شد. برای مثال مقادیر مثبت قبل از بسته شدن برای هر کانال مقدار ماکزیممی دارد و شروع بازه را جایی در نظر گرفتم که سیگنال به نصف مقدار اخیر رسیده باشد. برای مقادیر منفی هم وقتی از مینیمم منفی به نصف آن می‌رسیدم آنجا را انتهای بازه در نظر گرفتم. سپس برای هر کانال میانگین این مقادیر را محاسبه کردم. با توجه به نزدیک بودن آنها و این حدس که احتمالاً چند درصد خطا در بدست آوردن این بازه نباید تاثیری در نتایج نهایی بگذارد، از وقت گذاشتن و بدست آوردن بازه‌ها برای بقیه کانال‌ها صرف‌نظر نمودم. همچنین برای بدست آوردن بازه سیگنال باز شدن نیز با توجه به تجربه‌ای که در بالا اشاره شد فقط برای کانال ۱ بازه‌ها را بدست آوردم و سپس میانگین گرفتم. نحوه تغییرات برای سیگنال باز شدن نیز مشابه بسته شدن بود. بازه‌ها برای بسته شدن در جدول ۶ و برای باز شدن در جدول ۷ مشاهده می‌شوند.

Channel 1													Avg
Eye Closing No.	1	2	3	4	5	6	7	8	9	10	11	12	
Before(ms)	265	273	273	289	281	352	250	289	281	281	*	320	287
After(ms)	750	703	305	*	922	773	578	953	*	*	*	*	712
Channel 2													Avg
Eye Closing No.	1	2	3	4	5	6	7	8	9	10	11	12	
Before(ms)	265	273	273	289	273	336	242	305	273	266	*	312	282
After(ms)	750	703	305	*	1547	914	1547	*	*	*	*	*	961
Channel 3													Avg
Eye Closing No.	1	2	3	4	5	6	7	8	9	10	11	12	
Before(ms)	265	273	273	289	234	305	234	227	266	289	*	320	270
After(ms)	750	703	305	*	1375	1797	1687	656	*	*	*	*	1039
Channel 4													Avg
Eye Closing No.	1	2	3	4	5	6	7	8	9	10	11	12	
Before(ms)	265	273	273	289	215	359	234	273	266	281	*	312	276

After(ms)	750	703	305	*	422	891	1039	*	*	*	*	*	685
-----------	-----	-----	-----	---	-----	-----	------	---	---	---	---	---	-----

جدول ۶

Channel 1												Avg
Eye Opening No.	1	2	3	4	5	6	7	8	9	10	11	
Before(ms)	219	234	320	*	133	258	273	1741	195	297	414	408
After(ms)	844	336	422	*	547	656	344	547	594	820	250	536

جدول ۷

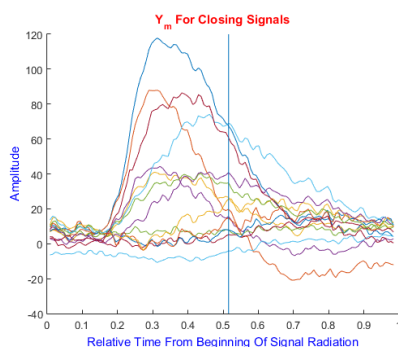
خانه‌هایی از جدول که با * مشخص شده‌اند یا رفتار گفته شده را از خود نشان نمی‌دادند یا بازه پلک زدن آنقدر کوتاه بوده که رفتار مورد نظر مشاهده نشده است.

بازه نهایی برای بسته شدن: ۵۱۶ میلی ثانیه قبل (۶۶ داده) تا ۴۶۱ میلی ثانیه بعد (۵۹ داده)

بازه نهایی برای باز شدن: ۵۳۹ میلی ثانیه قبل (۶۹ داده) تا ۲۵۰ میلی ثانیه بعد (۳۲ داده)

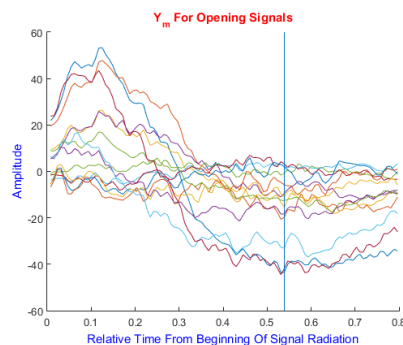
۳. الف) ابتدا آرایه‌ای ۴ بعدی تولید می‌شود با مقادیر صفر. بعد آخر مربوط به شماره کانال، بعد سوم مربوط به شماره دفعه بسته شدن پلک است. بعد دوم تعداد ستون‌های ماتریس سیگنالی است که می‌خواهیم بسازیم. بعد اول نیز سطرهای آن. (که به طول داده‌هاست که در بازه قسمت قبل قرار می‌گیرند). در ادامه سیگنال‌ها ساخته می‌شوند. چون آخرین بسته شدن پلک از آخر بازه تعداد کافی داده ندارد، برای آن سیگنال مربوط را جداگانه بدست می‌آوریم. سپس آرایه‌ای ۳ بعدی تولید می‌شود که سیگنال میانگین گرفته شده روی همه بسته شدن‌ها را برای هر کانال ذخیره کند و نهایتاً مقداردهی می‌شود.

ب) در این قسمت سیگنال‌هایی که در بالا بدست آمد بر روی یک شکل رسم شده‌اند. تصویر ۶ نتیجه را نشان می‌دهد. خط عمودی نیز زمان پلک زدن را نشان می‌دهد. (با دادن یک جفت نقطه رسم شد).



شکل ۶

۴. دقیقاً همان کارها انجام شد. با این تفاوت که آخرین باز شدن مشکل کم بودن تعداد داده نداشت. نتیجه در تصویر ۷ قابل مشاهده است.



شکل ۷

۵. برای هر کانال روی شکل بررسی شد و میانگین هر کدام در زیر آورده شده است.

برای بسته شدن: ۱۳۳ میلی ثانیه قبل

برای باز شدن: ۴۱۴ میلی ثانیه قبل

۶. الف) با استفاده از به توان ۲ رساندن انحراف معیار نتایج بدست آمده‌اند. (جدول ۸)

کانال	بسته شدن	باز شدن
۱	1590.72576506599	1162.03284443218
۲	1158.99417062514	528.354522501596
۳	147.930229942146	185.750062857792
۴	243.411267590985	150.986688450590
۵	2.78358316532509	3.20133904217612
۶	21.5110332951195	6.94007964571233
۷	20.0190439638972	10.6840561329761
۸	31.1180905587747	14.8966651364623
۹	35.2908862187733	14.2736580020962
۱۰	52.1323792194267	12.5802863392999
۱۱	148.287631552591	83.8229520183798
۱۲	110.081460400995	104.190493272199
۱۳	480.332454473274	246.534896750279
۱۴	917.915464227177	918.629699637406

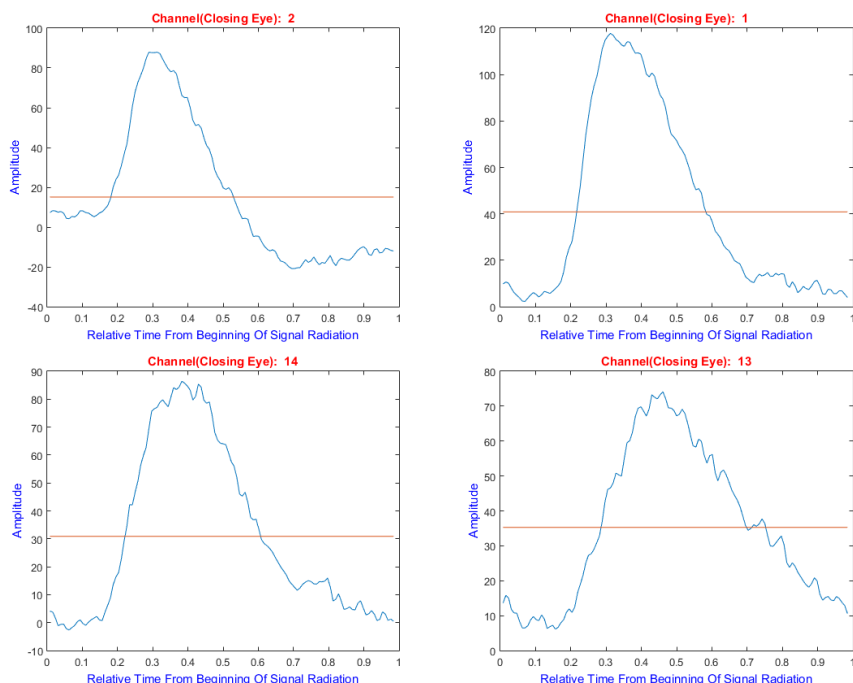
جدول ۸

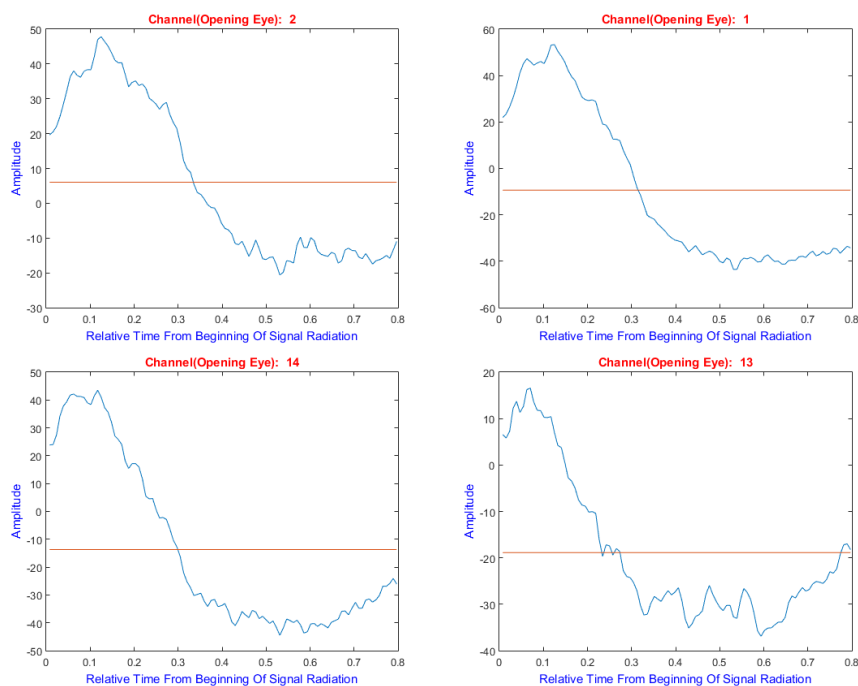
بیشترین واریانس مربوط به پروب‌های ۱ و ۲ و ۱۳ و ۱۴ می‌باشد. همانطور که در قسمت‌های بعدی اشاره شده است، زیاد بودن واریانس در یک سیگنال، یعنی سیگنال تغییرات دارد و با توجه به شناختی که تا اینجا

از سیگنال مغزی بدست آوردیم و داده‌های پرت آن را حذف نمودیم، مشخصاً این تغییرات زیاد، تغییرات بامعنا است و حاوی اطلاعات است. (ناشی از پراکندگی بیش از حد داده‌ها به خاطر داده پرت نیست).

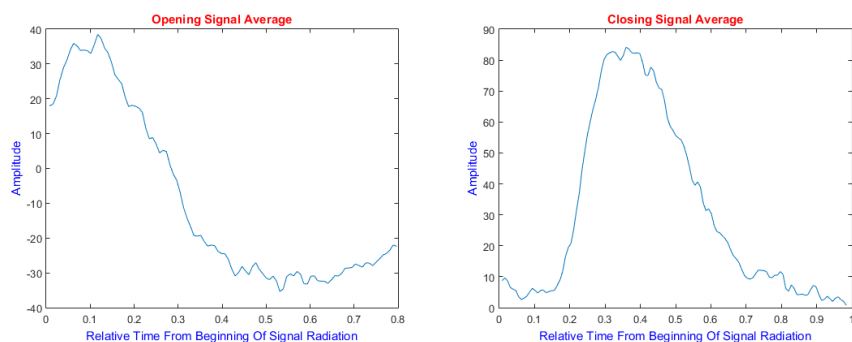
ب) برای هر دو حالت یکسان‌اند. (فقط ترتیب بزرگ بودن واریانس متفاوت است). این ۴ پروب همانطور که گفته شد ۱ و ۲ و ۱۳ و ۱۴ هستند. ۱ و ۲ سمت چپ سر و کنار چشم هستند. ۱۳ و ۱۴ متقارن آن‌ها در سمت راست هستند. (رجوع شود به شکل ۳ و ۴) همانطور که حدس زده می‌شد پروب‌های نزدیک تر به چشم، اطلاعات بسیار بیشتری دارند. (واریانس بیشتری دارند).

پ) نتایج در تصاویر زیر دیده می‌شود. نسبی در محور ایکس به معنی تقسیم نمی‌باشد. یعنی مبدا زمان، شروع سیگنال مربوط به باز و بسته شدن است. (نه زمان داده‌گیری از دستگاه) خط نارنجی مقدار میانگین را نشان می‌دهد.





میانگین بسته شدن و باز شدن:

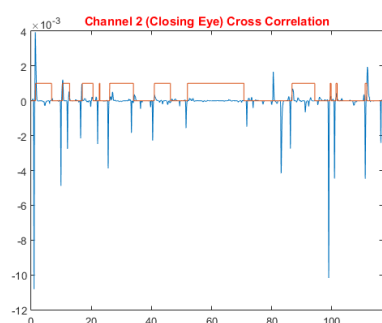


قسمت سوم. ساخت مدل تخمین

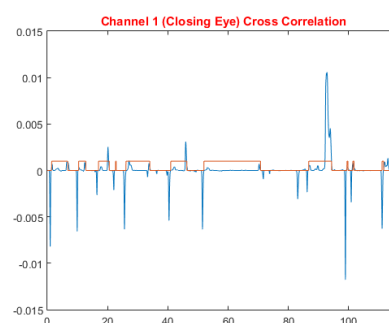
۱. نمودارهای خواسته شده در زیر قابل مشاهده‌اند. برای نرمالیزه بودن نتیجه کراس کوررلیشن از کلمه `coeff` به عنوان ورودی آخر تابع `xcorr` استفاده می‌کنیم. برای استفاده از آن باید بردارها هم اندازه باشند. همانطور که در کامنت‌گذاری گفته شده‌است، سیگنال‌های باز و بسته شدن `extend` شده‌اند با مقدار صفر تا بتوان عملیات مورد نظر را انجام داد. ضمناً برای رسم نمودارها، مقدار `x` به توان ۳ رسیده است. این کار باعث حفظ علامت شده و پیک‌های مهم را قابل تشخیص‌تر می‌کند و پیک‌های کوچک را حذف می‌کند. در صورت نیاز به مشاهده نتیجه اصلی باید در کد، توان ۳ برداشته شده و داده‌های مربوط به پلک زدن نیز در ضریب مناسب دیگری ضرب شوند.

* به دلیلی که متوجه نشدم نتیجه کراس کوررلیشن من نسبت به مبدا قرینه شده بود. برای همین در ادامه هر جا نیاز به رسم بوده‌است، نتیجه کراس کوررلیشن را نسبت به مبدا قرینه کرده‌ام. ($-x, -y$)

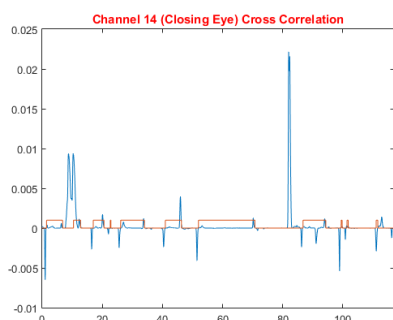
نتیجه: در شکل ۸، در لحظات بسته شدن ابتدا پیک منفی و سپس پیک مثبت داریم. قبل از باز شدن نیز پیک مثبت داریم. در شکل ۹ الگویی به منظمی الگوی قبل مشاهده نمی‌شود. در شکل ۱۰ قبل از بسته شدن پیک منفی داریم و قبل از باز شدن نیز پیک مثبت. البته نامنظمی‌هایی دیده می‌شود. الگوی شکل ۱۱ نیز مانند ۱۰ است با این تفاوت که بی‌نظمی بیشتری دارد. شکل ۱۲ در لحظات بسته شدن پیک مثبت و منفی پشت سرهم دارد و قبل از باز شدن پیک منفی. نامنظمی کمی دارد. شکل ۱۳ نامنظمی‌های زیادی دارد و الگوی مناسبی ندارد. شکل ۱۴ قبل از بسته شدن پیک مثبت و قبل از باز شدن پیک منفی دارد اما مقداری نامنظمی دارد. شکل ۱۵ نیز مانند شکل ۱۴ است با نامنظمی‌های بیشتر.



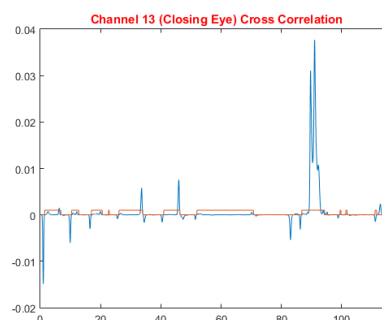
شکل ۹



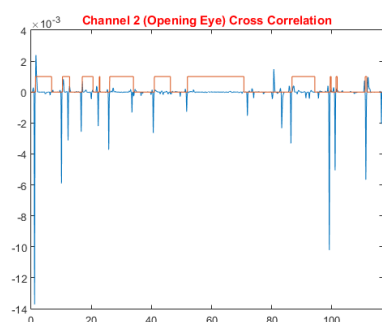
شکل ۸



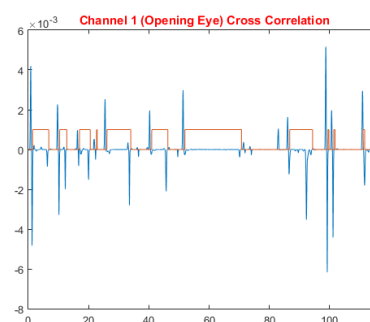
شکل ۱۱



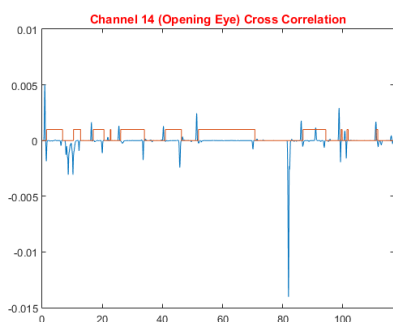
شکل ۱۰



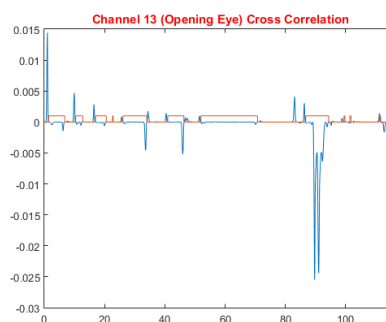
شکل ۱۳



شکل ۱۲



شکل ۱۵



شکل ۱۴

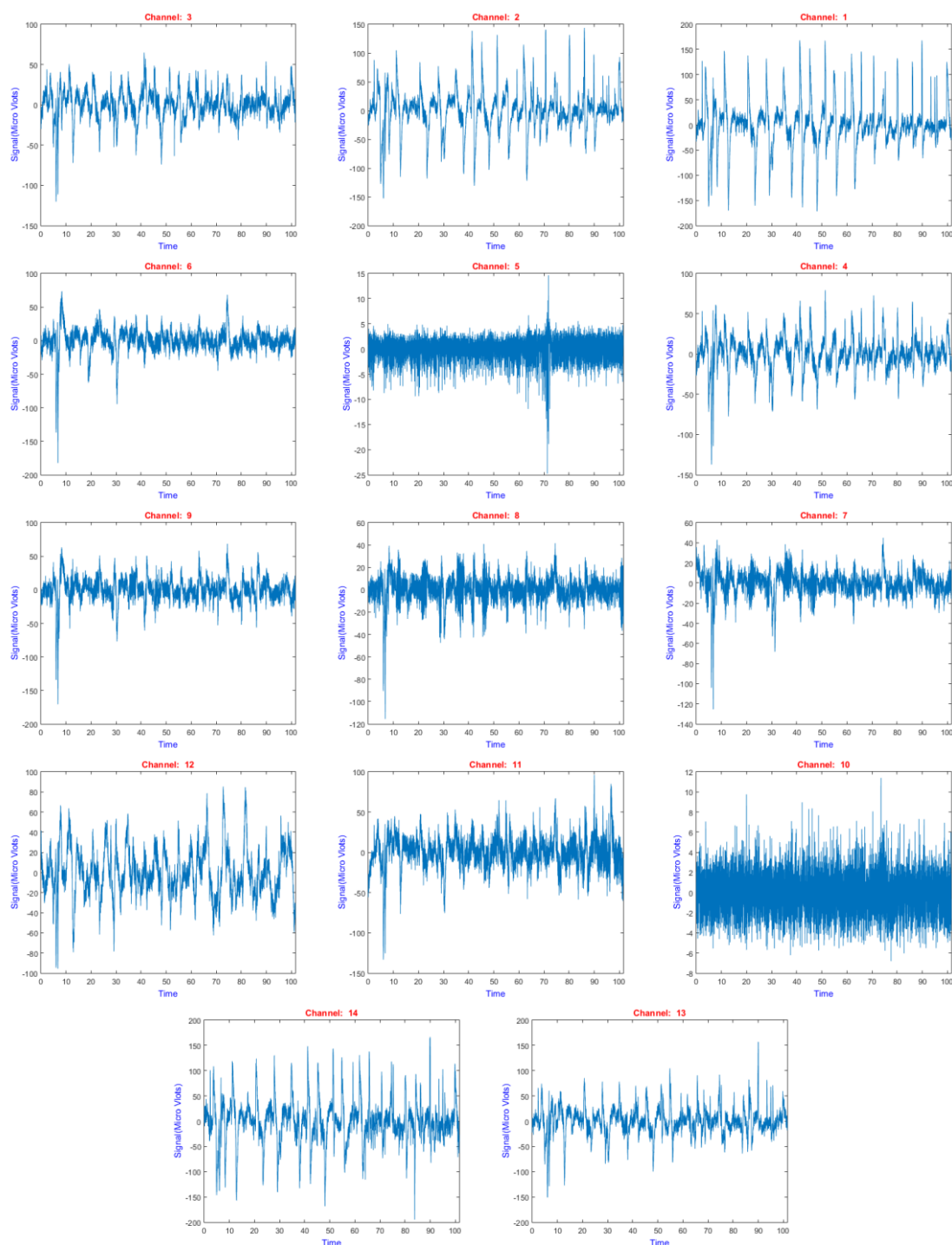
۲. با توجه به توضیحات اخیر، کوررلیشن سیگنال باز شدن با کانال ۱ اطلاعات دقیق و مطلوبی می‌دهد. کانال‌های دیگر نامنظمی‌های زیادی دارند. اما کانال اخیر الگوی منظمی دارد که در ادامه به آن می‌پردازیم. بنابراین از این به بعد تنها با سیگنال باز شدن و کانال ۱ کار می‌کنیم.

برای هر دسته از اطلاعات گرفته شده از دستگاه، ابتدا با روش‌های مشابه این قسمت داده‌های پرت را حذف می‌کنیم. از آزمایش Train برای همیشه سیگنال باز شدن را بدست آوردیم. از این سیگنال برای هر آزمایش دیگر می‌توانیم استفاده کنیم. پس ابتدا با توجه به طول سیگنال آن را extend می‌کنیم. سپس کوررلیشن نرمالیزه شده آن را با کانال ۱ سیگنال مغزی می‌گیریم. مقدار رگرسیون به توان ۳ را رسم می‌کنیم. الگوریتم: ابتدا پیک‌های مثبت و منفی را می‌یابیم. (توضیح روش در قسمت چهارم) اگر پیک مثبت و منفی با فاصله اندک (از داده‌های آزمایش Train بدست می‌آید). داشته باشیم، قطعاً پس از تاخیری (از داده‌های آزمایش Train بدست می‌آید). بسته شدن پلک را خواهیم داشت. پیک‌های منفی: اگر با فاصله اندک که در بالا گفته شد از یک پیک مثبت باشند که اطلاعات جدیدی ندارند و همان بسته شدن را اعلام می‌کنند. اگر از این فاصله اخیر از ماکزیمم قبلشان دورتر باشند و بلافاصله بعدشان پیک مثبت نباشد، با تاخیری مشخص (از داده‌های آزمایش Train بدست می‌آید). باز شدن اتفاق می‌افتد. اگر دو پیک متوالی منفی داشته باشیم نیز باز شدن اتفاق می‌افتد. (با توجه به شکل ۱۲)

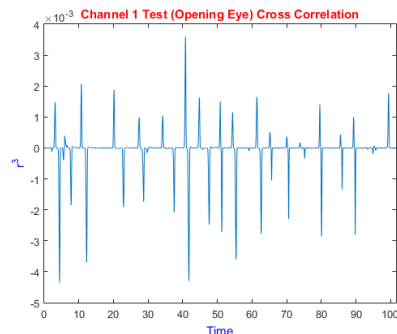
* اگر پلک زدن سریع داشته باشیم، سیگنال باز شدن فرصت ساطع شدن پیدا نمی‌کند. بنابراین متوالیا سیگنال‌های بسته شدن می‌بینیم. پس متوجه می‌شویم در این بازه حتماً باز شدن رخ داده است. متأسفانه روشی برای بدست آوردن زمان دقیق باز شدن بدست نیامد. اما با میانگین گرفتن از مدت زمانی که پس از پیک منفی مربوط به بسته شدن پلک‌زدن‌های سریع، باز شدن در آزمایش Train اتفاق می‌افتد، می‌توان زمان تقریبی باز شدن را برای پلک‌زدن‌های سریع بدست آورد. من نیز از همین روش استفاده کردم. این الگو برای آزمایش Train کار کرد و لذا برای ادامه نیز از همین الگو استفاده می‌شود.

قسمت چهارم. یافتن زمان پلک زدن‌ها

۱. ابتدا همه کانال‌ها را رسم می‌کنیم تا به صورت چشمی بررسی کنیم که آیا داده پرت وجود دارد یا خیر. همچنین فایل جدیدی می‌سازیم تا تغییرات روی سیگنال اصلی را در آن ذخیره کنیم. متغیر فایل ضمیمه را بدست می‌آوریم و فایل ساخته شده را برابر آن قرار می‌دهیم. برای هر ۱۴ کانال سیگنال را رسم می‌کنیم. (گرچه ما فقط از کانال ۱ استفاده می‌کنیم). نتیجه در تصاویر زیر نشان می‌دهد داده پرت (فاحش و خطاسازی) در سیگنال‌ها نیست و همه در محدوده معقول شناخته شده برای سیگنال‌های مغزی هستند.



در ادامه کد، سیگنال باز شدن با توجه به طول سیگنال جدید مغزی مربوط به فایل extend، Test می‌شود. کراس کوررلیشن نرمالیزه شده آن با سیگنال اصلی کانال ۱ رسم می‌شود. (رگرسیون به توان ۳ رسیده است). (شکل ۱۶)



شکل ۱۶

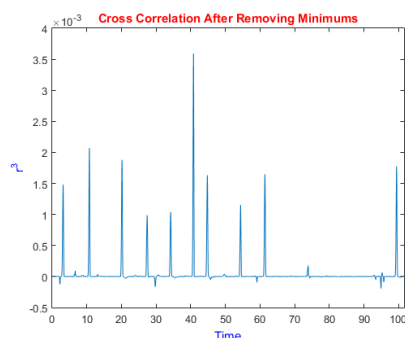
سپس به اجرای الگوریتم توضیح داده شده در قسمت قبل می‌پردازیم. حدی که از آن به بعد را پیک مثبت می‌نامیم، با بررسی الگو روی آزمایش Train و یافتن حد مناسب برای کارکرد الگو یافت شد. این مقدار برای وقتی که رگرسیون را به توان ۳ می‌رسانیم 35×10^{-5} بود. در ادامه با توجه به نیاز به عملیات بر روی رگرسیون، مشکل قرینه بودن نسبت به مبدا را با یک دستور در حلقه for حل کرده‌ام. سپس رگرسیون به توان ۳ می‌رسد. بعد از آن ماکزیمم رگرسیون یافت می‌شود. اگر از حد تعیین شده بیشتر باشد، ۱۰۰ داده قبل و بعد آن را صفر می‌کنم تا داده‌های نزدیک این ماکزیمم از بین بروند و ماکزیمم بعدی را بیابم. تعداد ۱۰۰ داده با آزمون و خطا بدست آمد. اندیس این ماکزیمم ها را در برداری ذخیره می‌کنیم. در تصویر ۱۷ نشان داده می‌شود که ماکزیمم‌ها پیدا شده و حذف شده‌اند. در نهایت به صورت صعودی مرتب می‌شوند. (sort)



شکل ۱۷

حدی که از آن به بعد را پیک منفی می‌نامیم، با همان روش یافتن حد پیک مثبت بدست آمد. (تطبیق الگو با آزمایش Train) این مقدار برای وقتی که رگرسیون را به توان ۳ می‌رسانیم -30×10^{-5} بود. در ادامه داده‌های رگرسیون که دستخوش تغییر شده بود، مجدداً ساخته می‌شوند و به توان ۳ می‌رسند. بعد از آن مینیمم رگرسیون یافت می‌شود. اگر از حد تعیین شده کمتر باشد، ۱۰۰ داده قبل و بعد آن را صفر می‌کنم تا داده‌های نزدیک این مینیمم از بین بروند و مینیمم بعدی را بیابم. تعداد ۱۰۰ داده با آزمون و خطا بدست

آمد. اندیس این مینیمم ها را در برداری ذخیره می کنیم. در تصویر ۱۸ نشان داده می شود که مینیمم ها پیدا شده و حذف شده اند. در نهایت به صورت صعودی مرتب می شوند. (sort)



شکل ۱۸

پس عملیات فوق که پیک های مثبت و منفی بدست آمد، باید با توجه به الگو زمان های بسته شدن و باز شدن را بیابیم.

بسته شدن: اگر در فاصله کمتر از $1/6$ ثانیه پس از یک پیک مثبت، پیک منفی داشته باشیم، قطعاً پس از $0/665$ ثانیه پلک بسته می شود. هر دوی این اعداد با بررسی همین الگو در آزمایش Train بدست آمده اند. به این صورت که برای هر بسته شدن فاصله پیک مثبت و منفی اندازه گیری شد و میانگین آنها به عنوان ماکزیمم فاصله تشخیص بسته شدن در نظر گرفته شد. همچنین برای هر بسته شدن، فاصله پیک مثبت تا بسته شدن پلک اندازه گیری شد و میانگین آنها در این قسمت لحاظ شد. (محاسبه دقیق این مقادیر با اندازه گیری های زیادی برای کانال ها انجام شد که آوردن آنها، گزارش را بیش از حد طولانی می کرد).

باز شدن: اولین مینیمم؛ آن قدر در حلقه while می مانیم تا نزدیک ترین ماکزیمم یافت شود. اگر از $1/6$ ثانیه که در قسمت قبل توضیح داده شد، با ماکزیمم قبلش فاصله داشت یعنی پس از $0/524$ ثانیه پلک باز می شود. این عدد نیز با میانگین گیری فاصله این گونه باز شدن ها تا باز شدن پلک (در آزمایش Train) بدست آمد.

مینیمم های بعدی: نزدیک ترین ماکزیمم به آنها پیدا می شود. اگر از $1/6$ ثانیه بیشتر فاصله داشتند و پس از مینیمم، در کمتر از $0/8$ ثانیه پیک مثبتی نباشد، (عدد بدست آمده مانند قبل) پس از $0/524$ ثانیه بعد پلک باز می شود. اگر پس از مینیمم، در کمتر از $0/8$ ثانیه پیک مثبتی باشد، باز شدن نخواهیم داشت. اگر با ماکزیمم قبلی از $1/6$ ثانیه بیشتر فاصله داشتند و دو پیک منفی متوالی داشته باشیم، پس از $0/524$ ثانیه از پیک دوم باز شدن داریم.

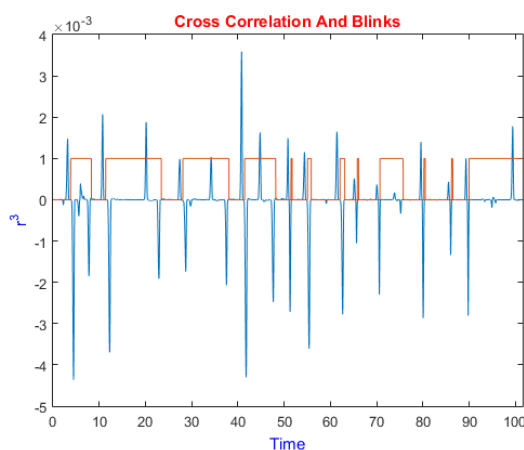
پلک زدن های سریع: اگر پس از یافتن پیک های مثبت و منفی پشت سرهم، پس از پیک منفی اولین بسته شدن، پیک مثبت و منفی پشت سرهم دیگری یافت شود، یعنی در این بازه باز شدنی وجود داشته که فرصت انتشار سیگنال خود را نداشته است. پس باید پس از مقداری که تقریبی است و با استفاده از پلک های سریع

آزمایش تمرینی بدست آمد، باز شدن اتفاق بیفتد. این مقدار ۰/۴۶۰ ثانیه بدست آمد. در نهایت این مقادیر باز شدن به ترتیب صعودی مرتب می‌شوند.

تولید آرایه Result

فایل Result.mat ساخته شده و در مموری لود می‌شود. بردار نهایی ابتدا به طول سیگنال ساخته شده و مقدار صفر می‌گیرد. نشانگر بسته بودن در ابتدا صفر است. به ازای هر داده از این آرایه نهایی وارد حلقه for می‌شود. اگر نشانگر بسته بودن صفر باشد، داده‌ها را صفر می‌کند تا برسد به اولین بسته شدن. در اینجا نشانگر بسته بودن یک شده و داده‌های بعدی وارد قسمت دیگر حلقه می‌شوند و عملیاتی مشابه صورت می‌گیرد. با نگاه به شکل کراس کوررلیشن مشخص است آخرین اتفاق بسته شدن بوده و بنابراین اگر به آخرین باز شدن برسیم، از آنجا تا آخر را باید یک نگه داریم و دیگر در جستجوی باز شدن نباشیم. قسمت آخر حلقه این مشکل را رفع می‌کند.

در آخر رگرسیون برای رسم نتیجه نهایی دوباره بازیابی شده و همراه زمان‌های باز و بسته شدن بدست آمده رسم می‌شود. (شکل ۱۹) نتایج بردار خواسته شده نیز به صورت ستونی در فایل Result ذخیره می‌شود.



شکل ۱۹