

Data acquisition, visualization and analysis in functional near-infrared spectroscopy (fNIRS)

Jan Dabrowski, 1765583

Individual Study 1

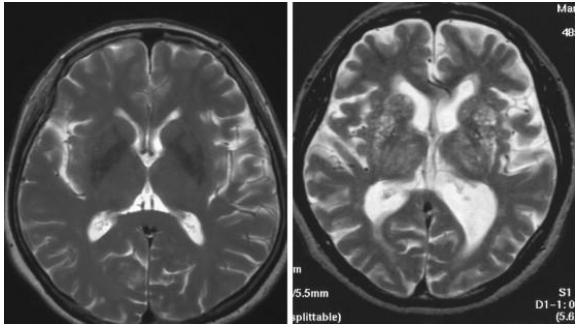
2019/20

Semester 1

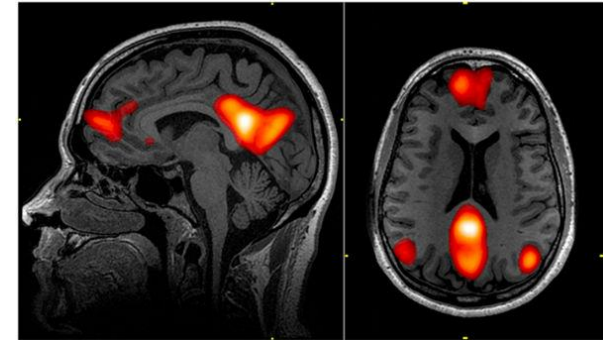
Supervisor: Prof. Hamid Dehghani

- 1. Brain Imaging**
- 2. fNIRS**
- 3. Building the solution**
- 4. Demo**
- 5. Data Mining**
- 6. The future**

Brain Imaging



Structural imaging



Functional imaging

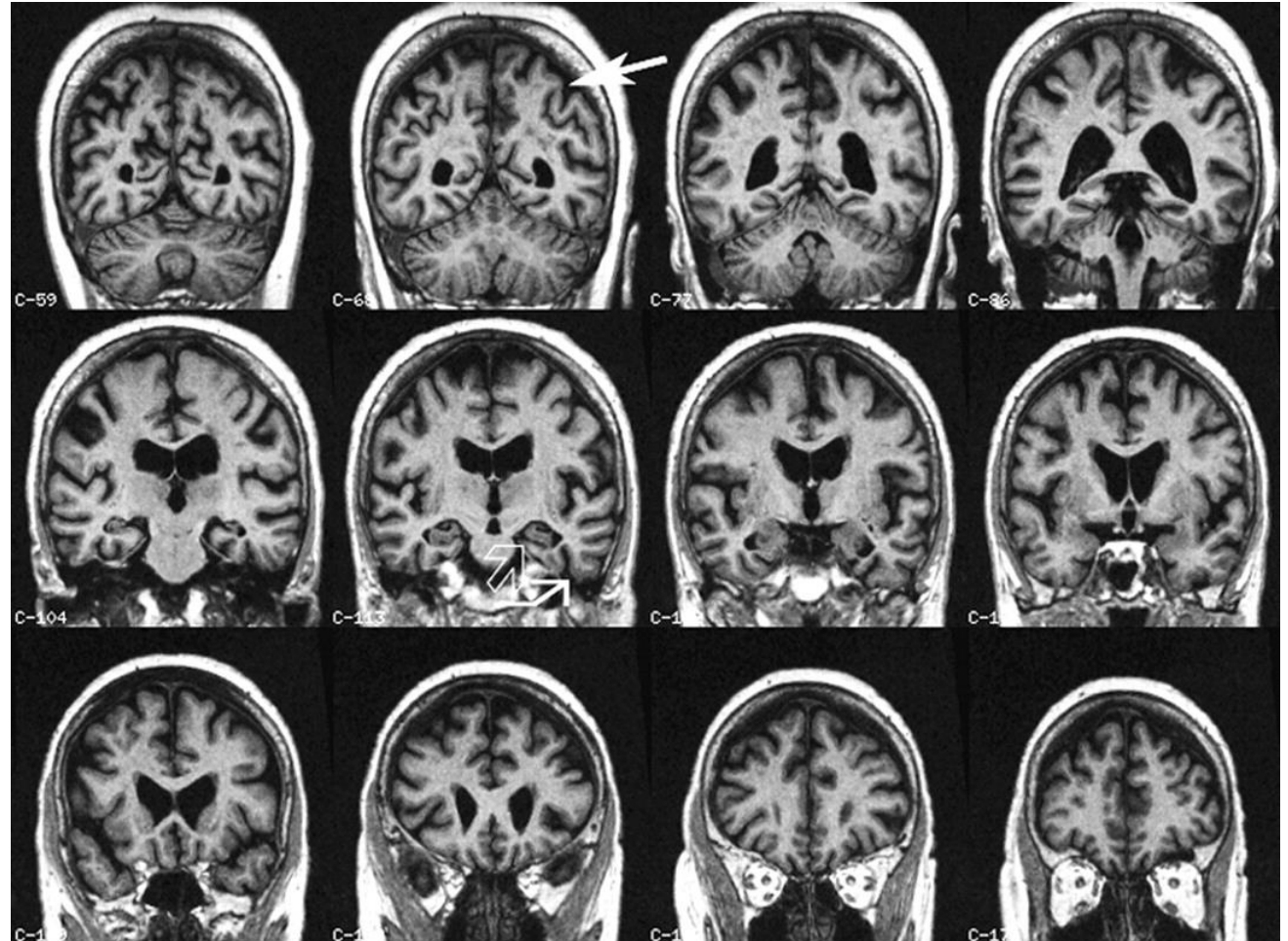
Brain Imaging

Structural imaging

Computed
tomography
(CT)

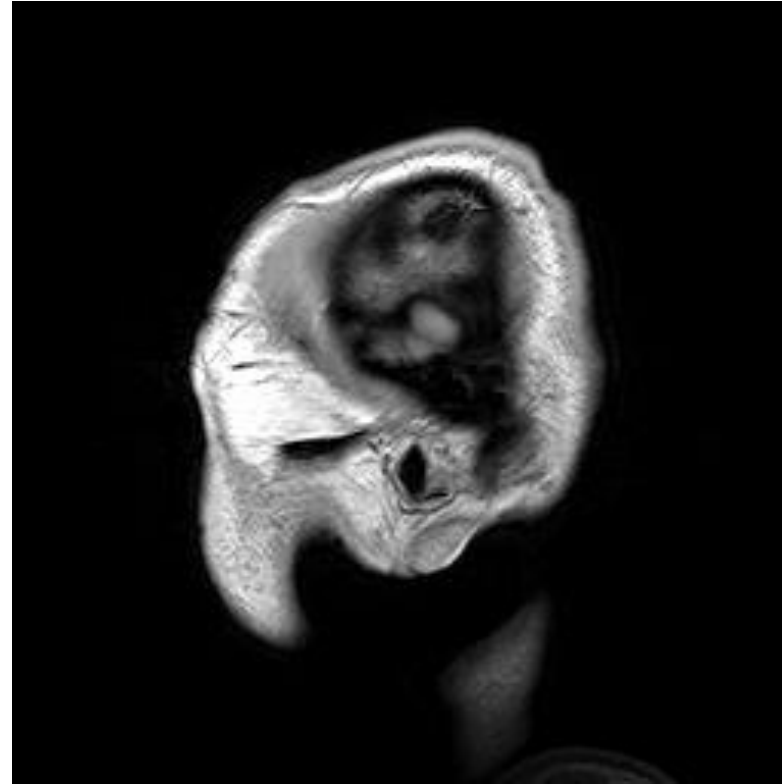
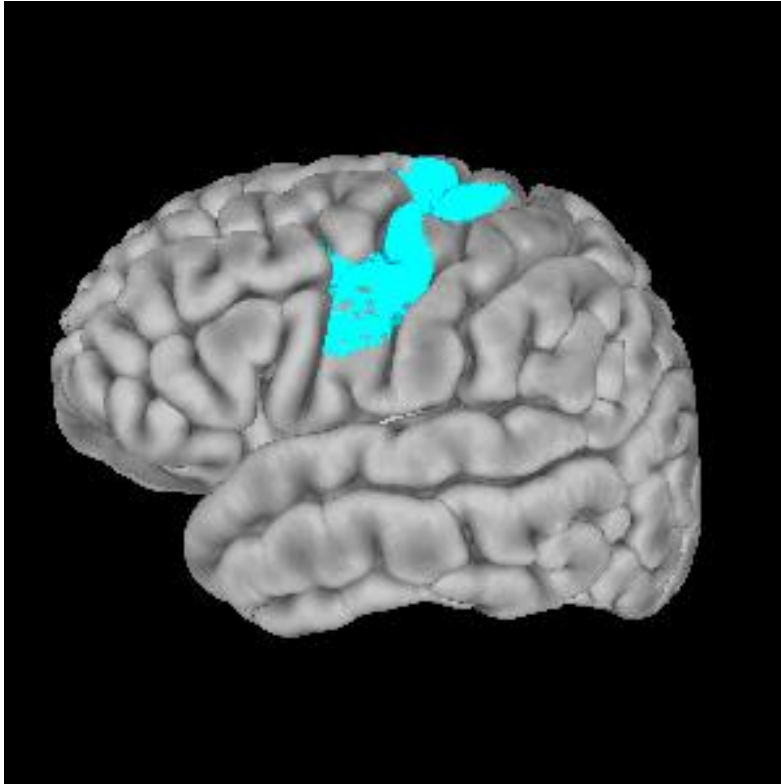


Magnetic
resonance
imaging
(MRI)



Brain Imaging

Structural imaging



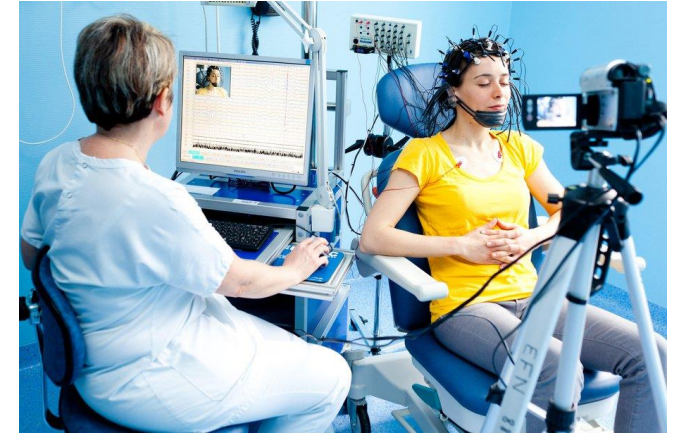
Brain Imaging

Functional imaging

Functional
magnetic
resonance
imaging
(fMRI)



Electro-
encephalography
(EEG)



Positron
Emission
Tomography
(PET)

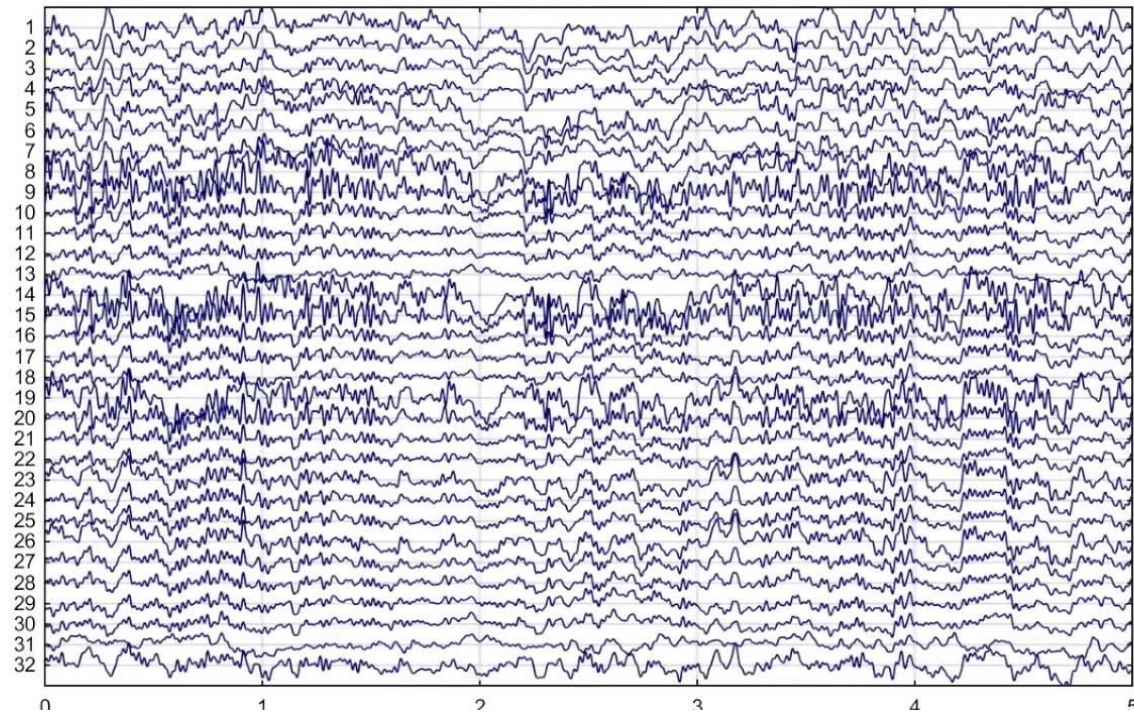


Magneto-
encephalography
(MEG)

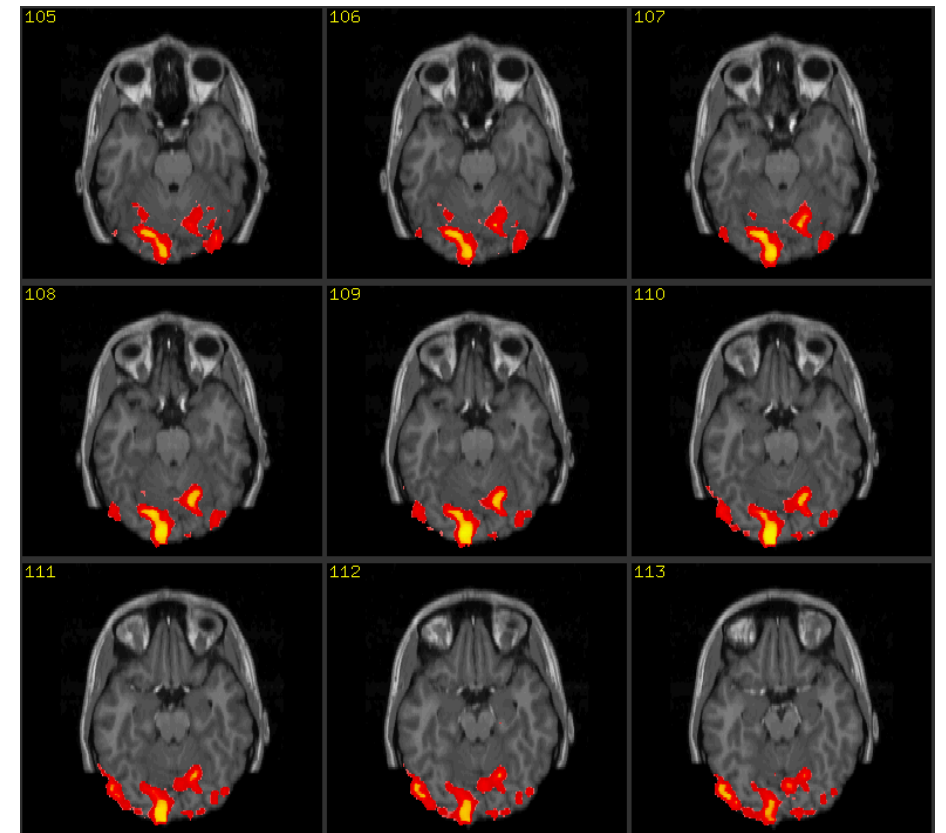


Brain Imaging

Functional imaging



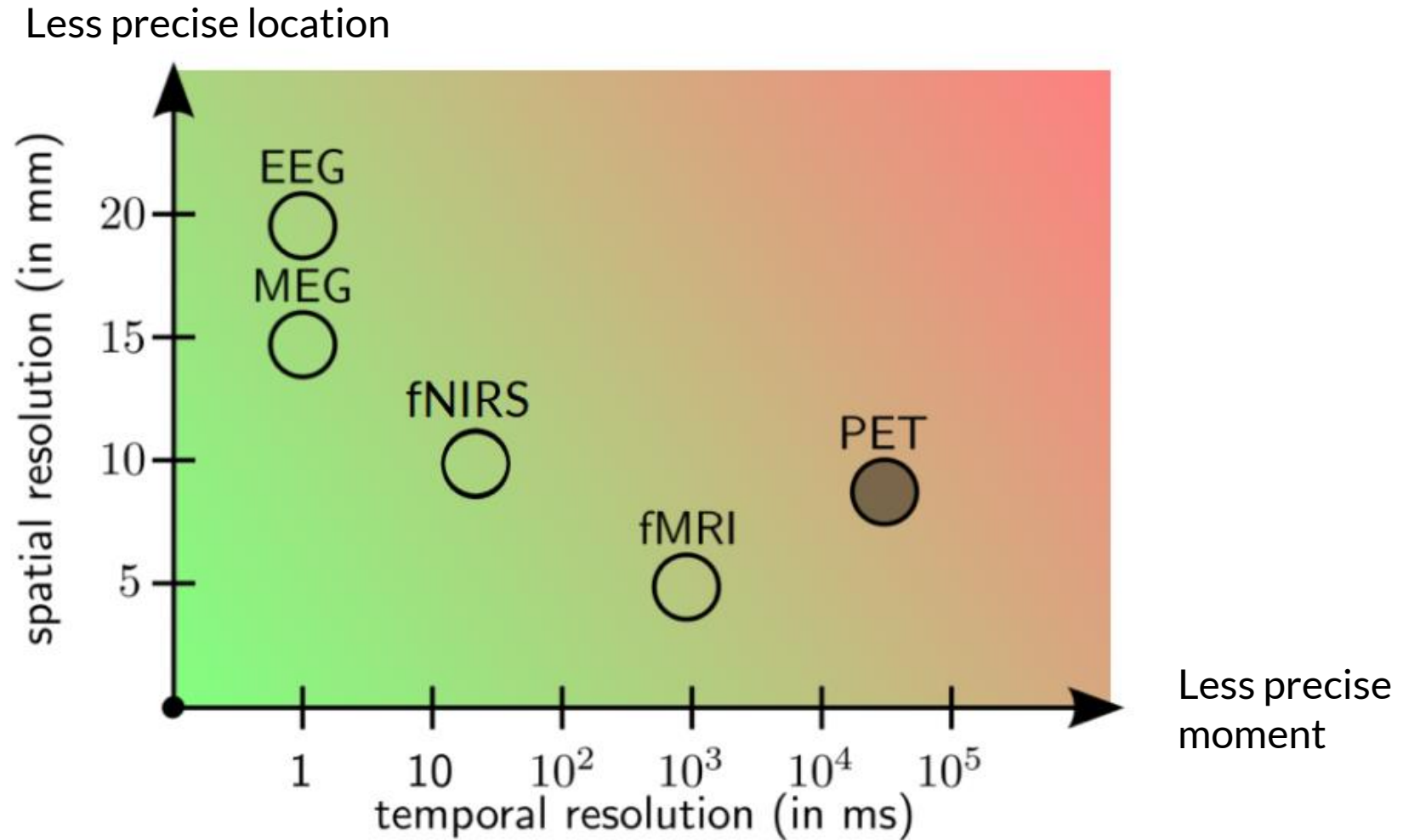
EEG data



fMRI data

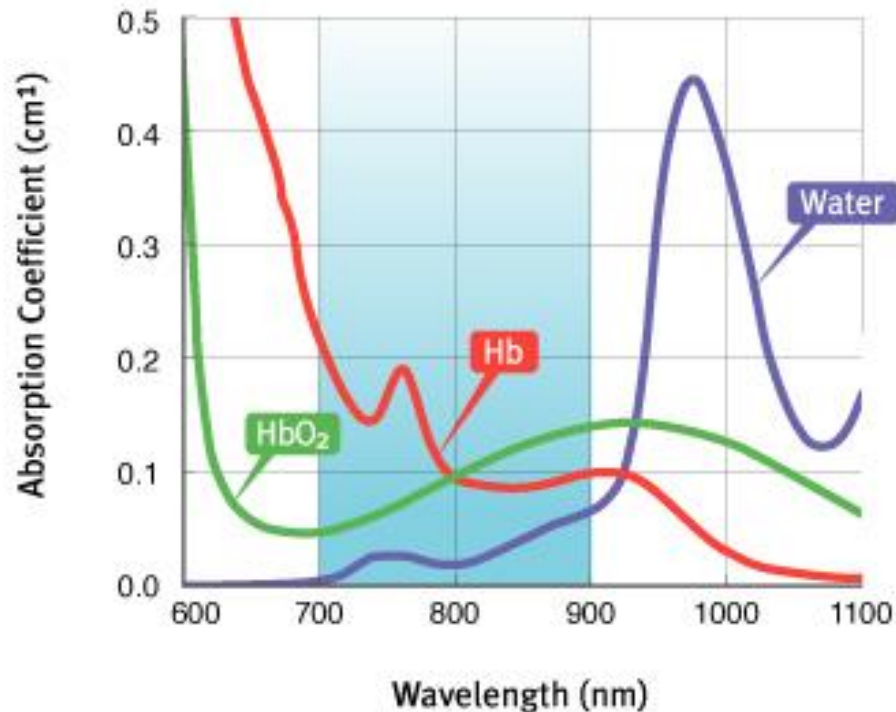
Brain Imaging

The future

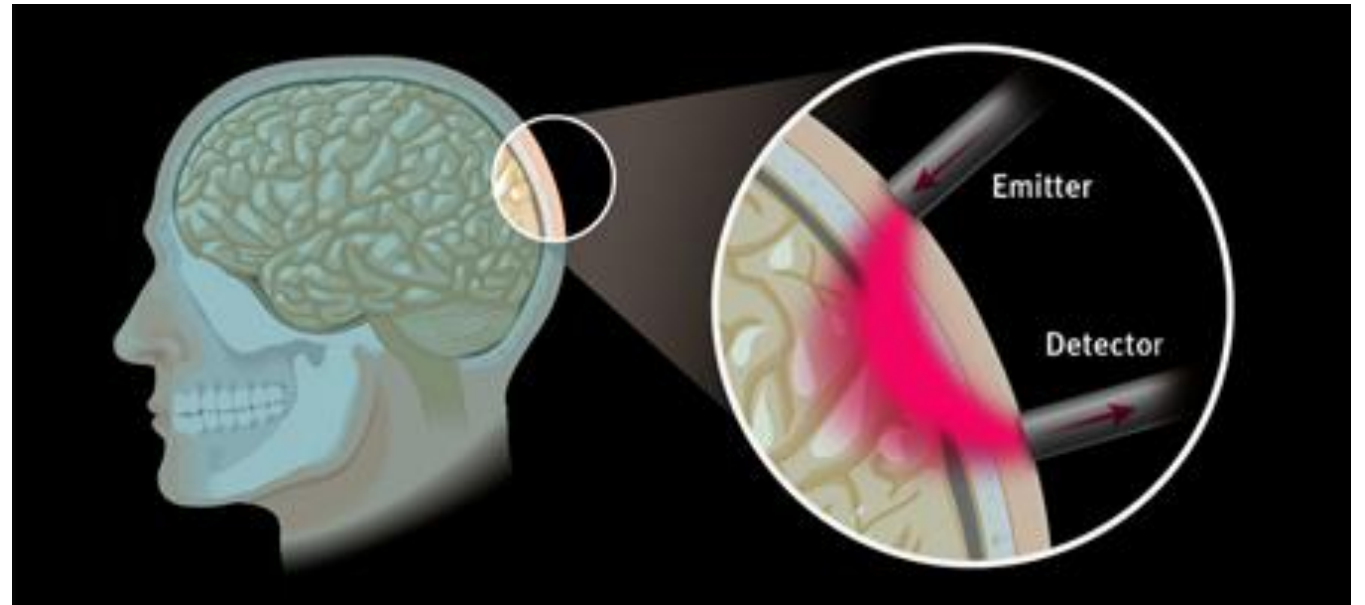


Functional near-infrared spectroscopy (fNIRS)

Main light absorbers

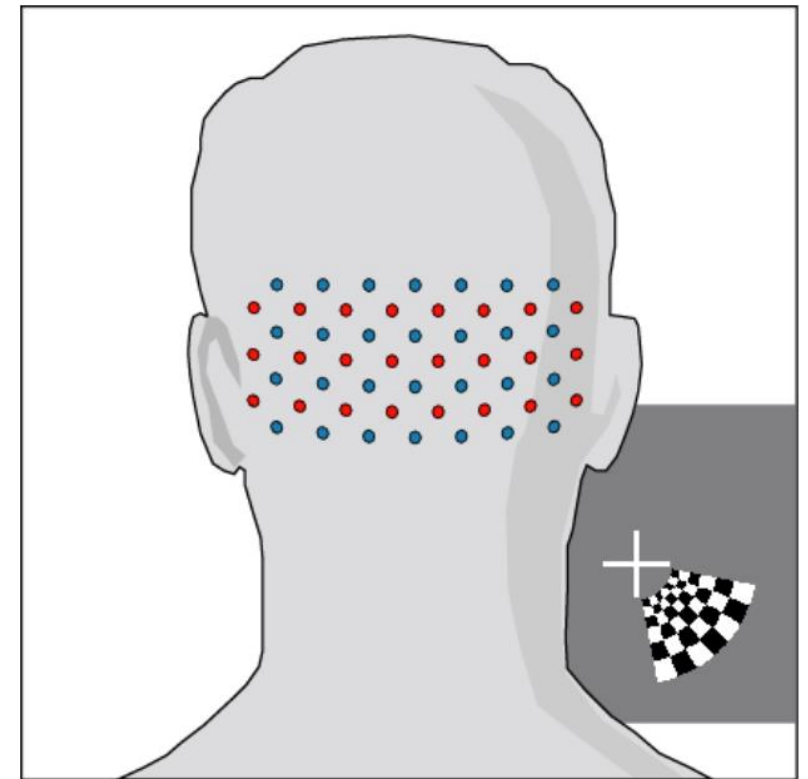
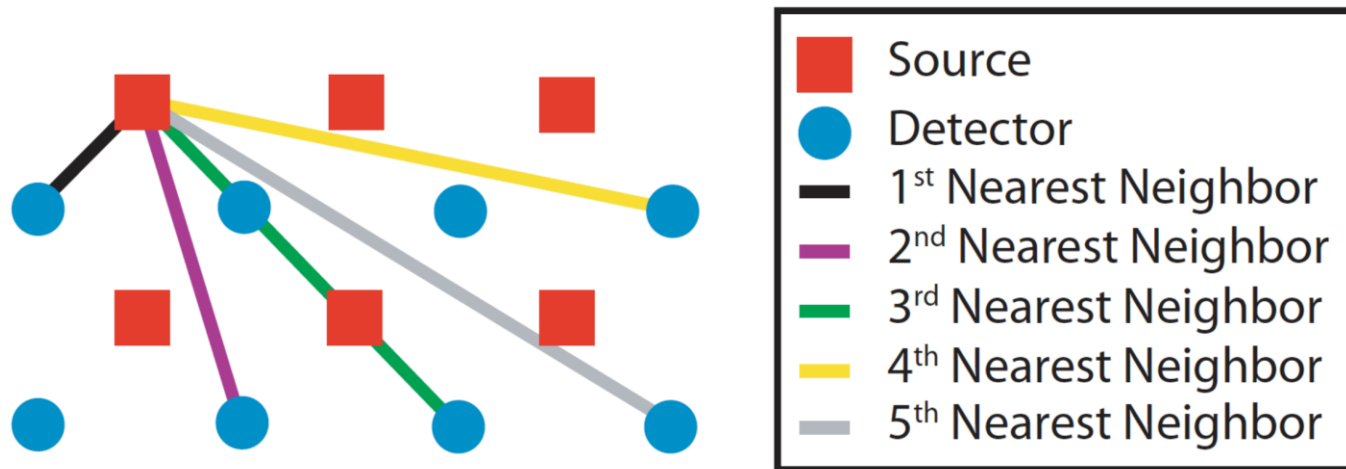


Light penetration in brain tissue

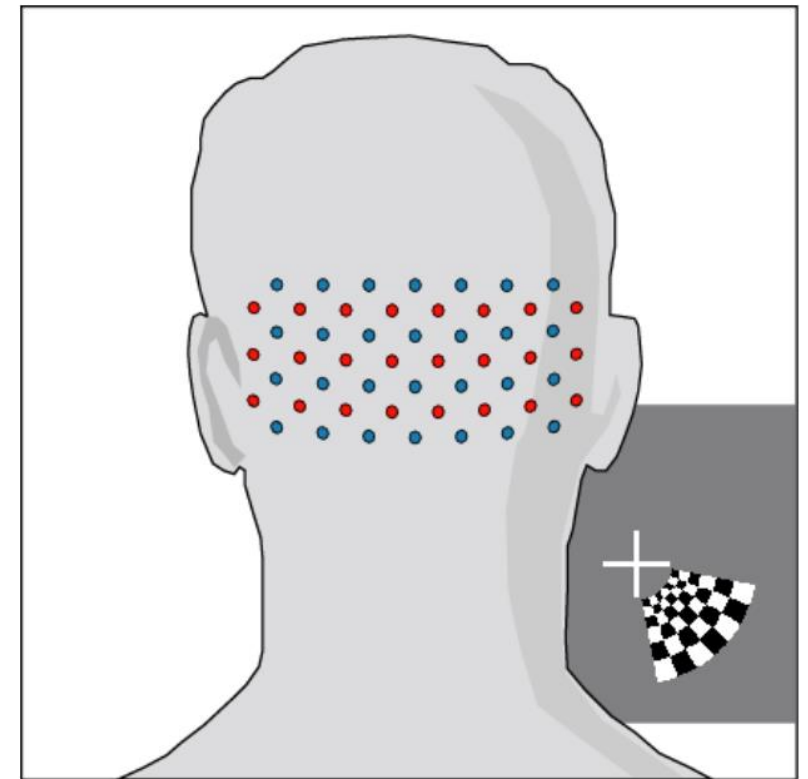
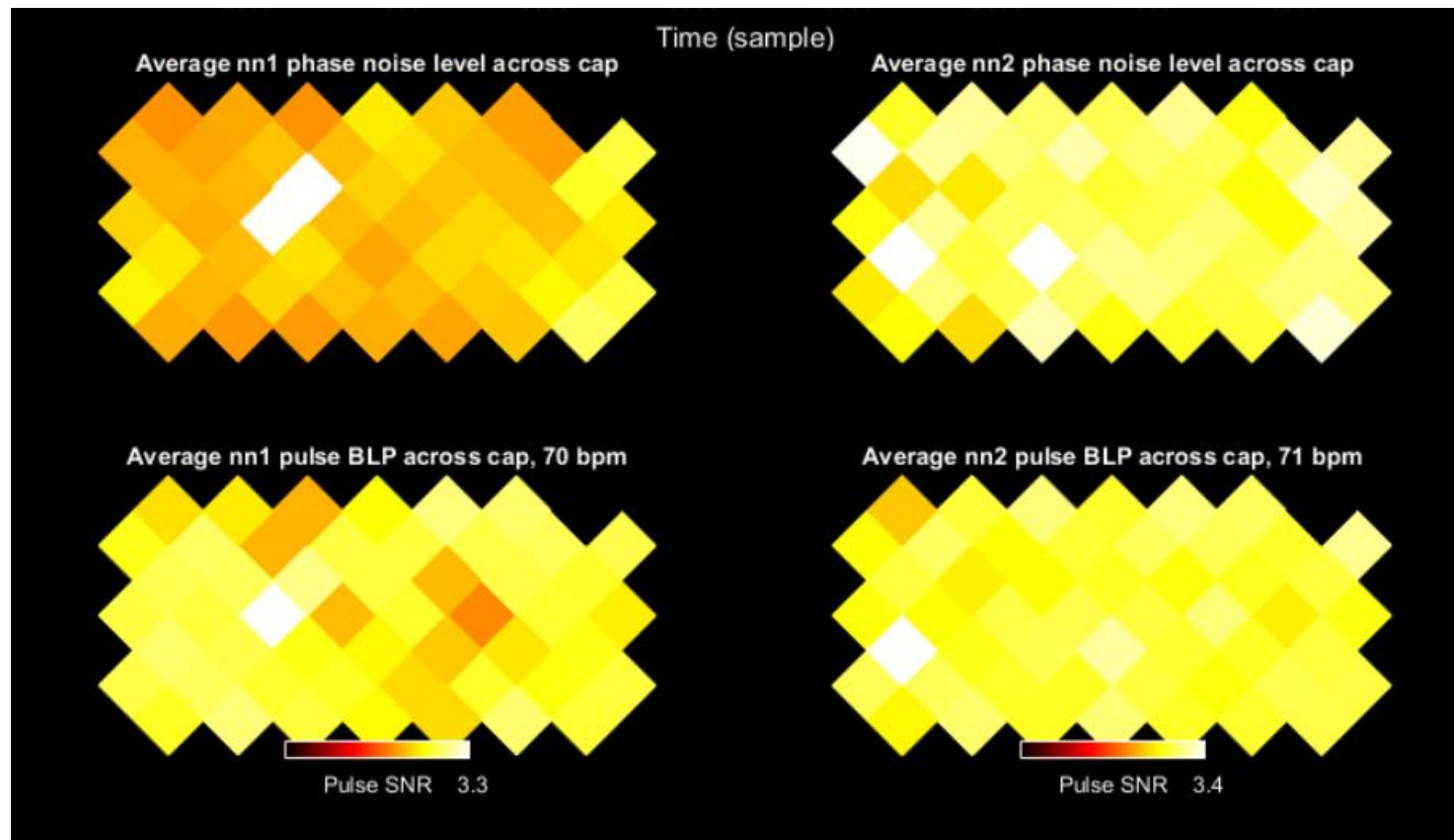


fNIRS

fNIRS source-detector layout



fNIRS

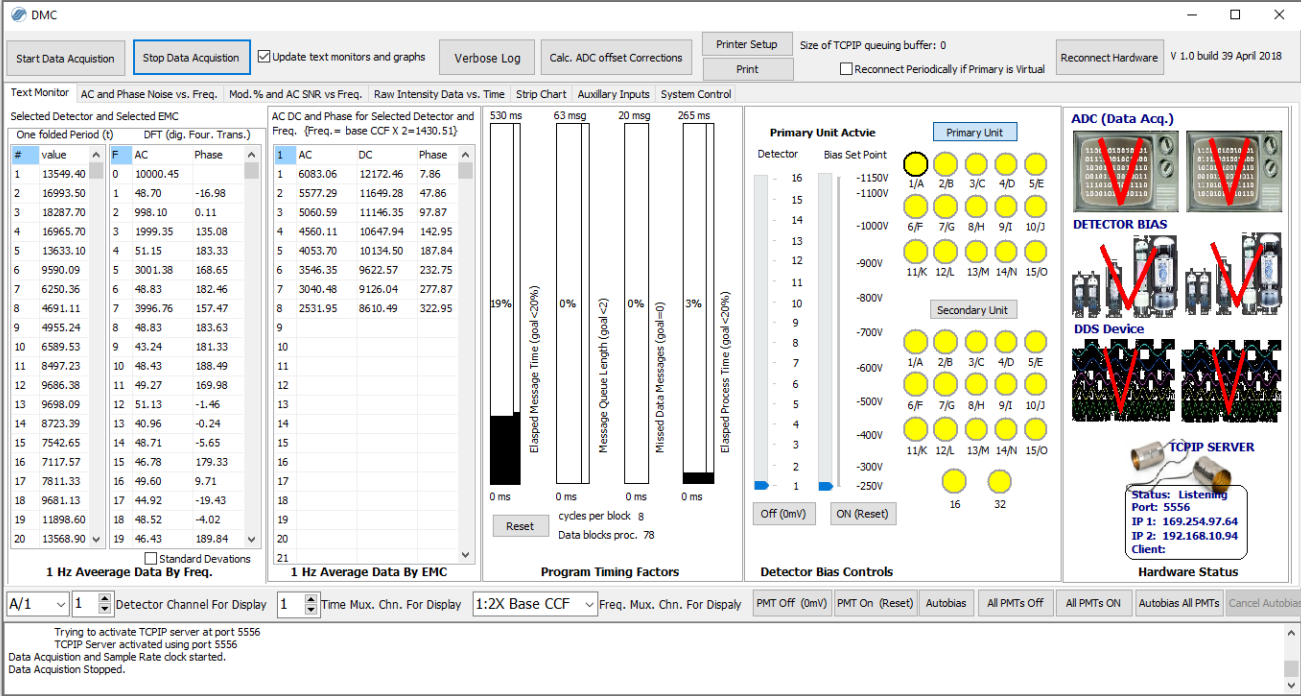
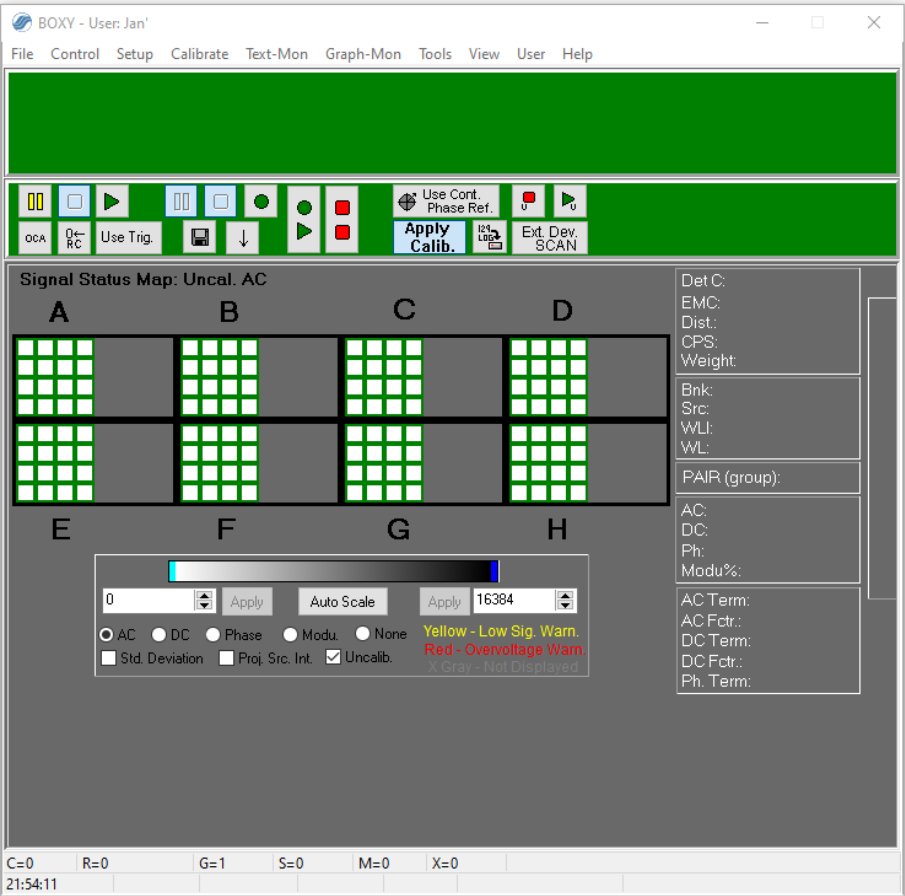


fNIRS



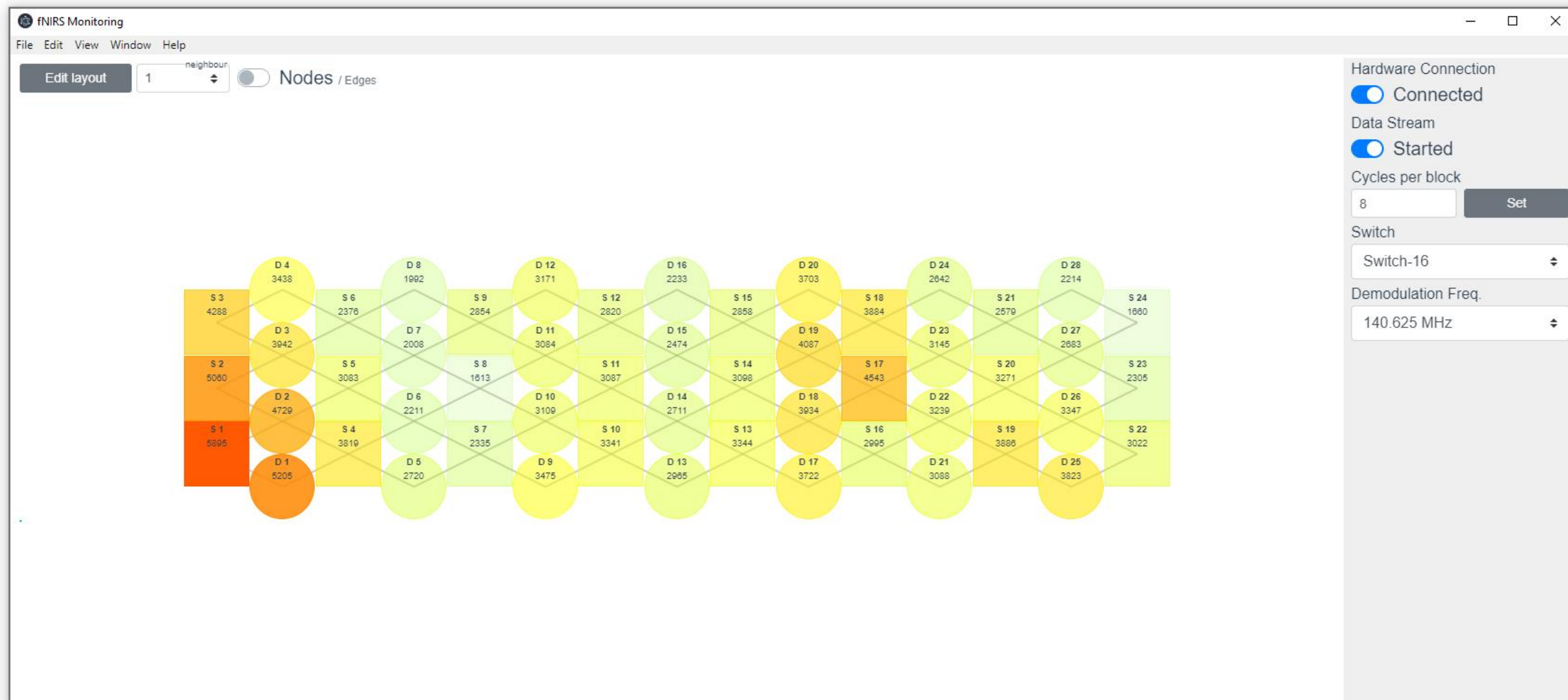
fNIRS

Available manufacturer's software



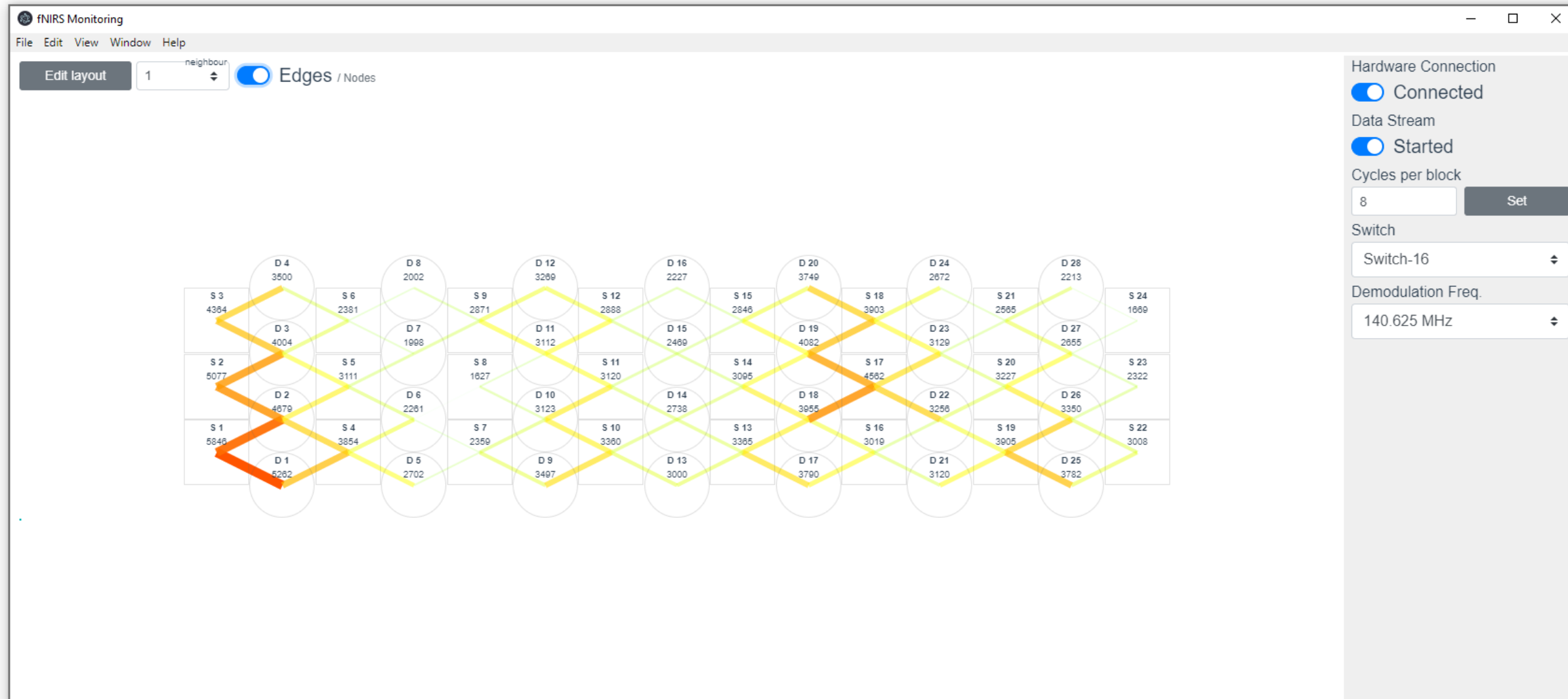
fNIRS

Final solution

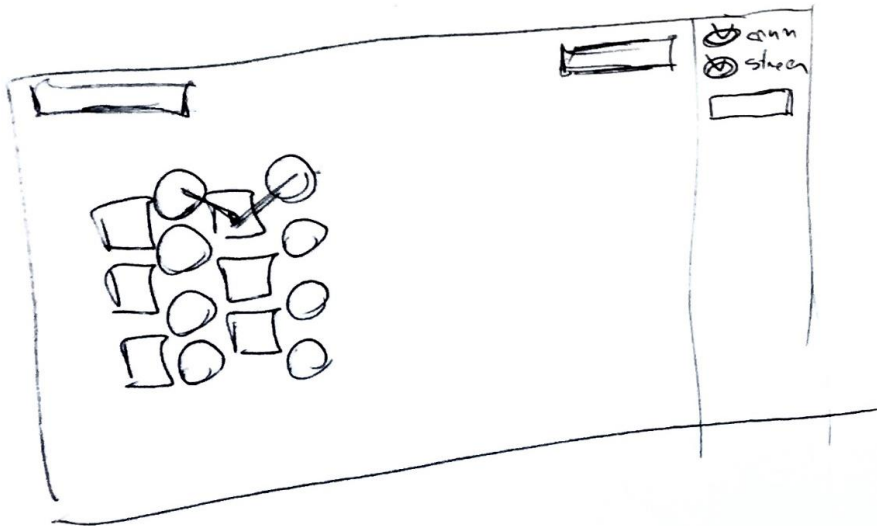


fNIRS

Final solution



Building the solution



```
Memory 176
Graph.cs 177
HubStore.cs M 178
> obj 180
> Properties 181
() appsettings.Development.js... 182
() appsettings.json M 183
  fnirs.csproj 184
  fnirs.csproj.user 185
  fnirs.log 186
  fnirs.sln 187
() global.json U 188
C new_ftmath.h 189
C Program.cs M 190
C Startup.cs M 191
C structs.h 192
> Backend.Tests 193
> data 194
> UI 195
  dist 196
  node_modules 197
  public 198
  src 199
    assets 200
    components 201
      graph 202
        Connections.vue 203
        NeighbourSelect.vue 204
        sidebar 205
          Connection.vue 206
          CyclesNum.vue M 207
          Frequency.vue M 208
          Streaming.vue M 209
          SwitchMode.vue M 210
          Board.vue M 211
          BoardEdit.vue M 212
          Layout.vue M 213
          SideBar.vue 214
        router 215
        store 216
        views 217
      App.vue M 218
  demolpc.js 219
  main.js 220
  SocketIpc.js 221
  tests 222
  browserconfig 223
  ... 224
  ... 225
  ... 226
  ... 227
  ... 228
  ... 229
  ... 230
  ... 231
  ... 232
  ... 233
  ... 234
  ... 235
  ... 236
  ... 237
  ... 238
  ... 239
  ... 240
  ... 241
  ... 242
  ... 243
  ... 244
  ... 245
  ... 246
  ... 247
  ... 248
  ... 249
  ... 250
  ... 251
  ... 252
  ... 253
  ... 254
  ... 255
  ... 256
  ... 257
  ... 258
  ... 259
  ... 260
  ... 261
  ... 262
  ... 263
  ... 264
  ... 265
  ... 266
  ... 267
  ... 268
  ... 269
  ... 270
  ... 271
  ... 272
  ... 273
  ... 274
  ... 275
  ... 276
  ... 277
  ... 278
  ... 279
  ... 280
  ... 281
  ... 282
  ... 283
  ... 284
  ... 285
  ... 286
  ... 287
  ... 288
  ... 289
  ... 290
  ... 291
  ... 292
  ... 293
  ... 294
  ... 295
  ... 296
  ... 297
  ... 298
  ... 299
  ... 300
  ... 301
  ... 302
  ... 303
  ... 304
  ... 305
  ... 306
  ... 307
  ... 308
  ... 309
  ... 310
  ... 311
  ... 312
  ... 313
  ... 314
  ... 315
  ... 316
  ... 317
  ... 318
  ... 319
  ... 320
  ... 321
  ... 322
  ... 323
  ... 324
  ... 325
  ... 326
  ... 327
  ... 328
  ... 329
  ... 330
  ... 331
  ... 332
  ... 333
  ... 334
  ... 335
  ... 336
  ... 337
  ... 338
  ... 339
  ... 340
  ... 341
  ... 342
  ... 343
  ... 344
  ... 345
  ... 346
  ... 347
  ... 348
  ... 349
  ... 350
  ... 351
  ... 352
  ... 353
  ... 354
  ... 355
  ... 356
  ... 357
  ... 358
  ... 359
  ... 360
  ... 361
  ... 362
  ... 363
  ... 364
  ... 365
  ... 366
  ... 367
  ... 368
  ... 369
  ... 370
  ... 371
  ... 372
  ... 373
  ... 374
  ... 375
  ... 376
  ... 377
  ... 378
  ... 379
  ... 380
  ... 381
  ... 382
  ... 383
  ... 384
  ... 385
  ... 386
  ... 387
  ... 388
  ... 389
  ... 390
  ... 391
  ... 392
  ... 393
  ... 394
  ... 395
  ... 396
  ... 397
  ... 398
  ... 399
  ... 400
  ... 401
  ... 402
  ... 403
  ... 404
  ... 405
  ... 406
  ... 407
  ... 408
  ... 409
  ... 410
  ... 411
  ... 412
  ... 413
  ... 414
  ... 415
  ... 416
  ... 417
  ... 418
  ... 419
  ... 420
  ... 421
  ... 422
  ... 423
  ... 424
  ... 425
  ... 426
  ... 427
  ... 428
  ... 429
  ... 430
  ... 431
  ... 432
  ... 433
  ... 434
  ... 435
  ... 436
  ... 437
  ... 438
  ... 439
  ... 440
  ... 441
  ... 442
  ... 443
  ... 444
  ... 445
  ... 446
  ... 447
  ... 448
  ... 449
  ... 450
  ... 451
  ... 452
  ... 453
  ... 454
  ... 455
  ... 456
  ... 457
  ... 458
  ... 459
  ... 460
  ... 461
  ... 462
  ... 463
  ... 464
  ... 465
  ... 466
  ... 467
  ... 468
  ... 469
  ... 470
  ... 471
  ... 472
  ... 473
  ... 474
  ... 475
  ... 476
  ... 477
  ... 478
  ... 479
  ... 480
  ... 481
  ... 482
  ... 483
  ... 484
  ... 485
  ... 486
  ... 487
  ... 488
  ... 489
  ... 490
  ... 491
  ... 492
  ... 493
  ... 494
  ... 495
  ... 496
  ... 497
  ... 498
  ... 499
  ... 500
  ... 501
  ... 502
  ... 503
  ... 504
  ... 505
  ... 506
  ... 507
  ... 508
  ... 509
  ... 510
  ... 511
  ... 512
  ... 513
  ... 514
  ... 515
  ... 516
  ... 517
  ... 518
  ... 519
  ... 520
  ... 521
  ... 522
  ... 523
  ... 524
  ... 525
  ... 526
  ... 527
  ... 528
  ... 529
  ... 530
  ... 531
  ... 532
  ... 533
  ... 534
  ... 535
  ... 536
  ... 537
  ... 538
  ... 539
  ... 540
  ... 541
  ... 542
  ... 543
  ... 544
  ... 545
  ... 546
  ... 547
  ... 548
  ... 549
  ... 550
  ... 551
  ... 552
  ... 553
  ... 554
  ... 555
  ... 556
  ... 557
  ... 558
  ... 559
  ... 560
  ... 561
  ... 562
  ... 563
  ... 564
  ... 565
  ... 566
  ... 567
  ... 568
  ... 569
  ... 570
  ... 571
  ... 572
  ... 573
  ... 574
  ... 575
  ... 576
  ... 577
  ... 578
  ... 579
  ... 580
  ... 581
  ... 582
  ... 583
  ... 584
  ... 585
  ... 586
  ... 587
  ... 588
  ... 589
  ... 590
  ... 591
  ... 592
  ... 593
  ... 594
  ... 595
  ... 596
  ... 597
  ... 598
  ... 599
  ... 600
  ... 601
  ... 602
  ... 603
  ... 604
  ... 605
  ... 606
  ... 607
  ... 608
  ... 609
  ... 610
  ... 611
  ... 612
  ... 613
  ... 614
  ... 615
  ... 616
  ... 617
  ... 618
  ... 619
  ... 620
  ... 621
  ... 622
  ... 623
  ... 624
  ... 625
  ... 626
  ... 627
  ... 628
  ... 629
  ... 630
  ... 631
  ... 632
  ... 633
  ... 634
  ... 635
  ... 636
  ... 637
  ... 638
  ... 639
  ... 640
  ... 641
  ... 642
  ... 643
  ... 644
  ... 645
  ... 646
  ... 647
  ... 648
  ... 649
  ... 650
  ... 651
  ... 652
  ... 653
  ... 654
  ... 655
  ... 656
  ... 657
  ... 658
  ... 659
  ... 660
  ... 661
  ... 662
  ... 663
  ... 664
  ... 665
  ... 666
  ... 667
  ... 668
  ... 669
  ... 670
  ... 671
  ... 672
  ... 673
  ... 674
  ... 675
  ... 676
  ... 677
  ... 678
  ... 679
  ... 680
  ... 681
  ... 682
  ... 683
  ... 684
  ... 685
  ... 686
  ... 687
  ... 688
  ... 689
  ... 690
  ... 691
  ... 692
  ... 693
  ... 694
  ... 695
  ... 696
  ... 697
  ... 698
  ... 699
  ... 700
  ... 701
  ... 702
  ... 703
  ... 704
  ... 705
  ... 706
  ... 707
  ... 708
  ... 709
  ... 710
  ... 711
  ... 712
  ... 713
  ... 714
  ... 715
  ... 716
  ... 717
  ... 718
  ... 719
  ... 720
  ... 721
  ... 722
  ... 723
  ... 724
  ... 725
  ... 726
  ... 727
  ... 728
  ... 729
  ... 730
  ... 731
  ... 732
  ... 733
  ... 734
  ... 735
  ... 736
  ... 737
  ... 738
  ... 739
  ... 740
  ... 741
  ... 742
  ... 743
  ... 744
  ... 745
  ... 746
  ... 747
  ... 748
  ... 749
  ... 750
  ... 751
  ... 752
  ... 753
  ... 754
  ... 755
  ... 756
  ... 757
  ... 758
  ... 759
  ... 760
  ... 761
  ... 762
  ... 763
  ... 764
  ... 765
  ... 766
  ... 767
  ... 768
  ... 769
  ... 770
  ... 771
  ... 772
  ... 773
  ... 774
  ... 775
  ... 776
  ... 777
  ... 778
  ... 779
  ... 780
  ... 781
  ... 782
  ... 783
  ... 784
  ... 785
  ... 786
  ... 787
  ... 788
  ... 789
  ... 790
  ... 791
  ... 792
  ... 793
  ... 794
  ... 795
  ... 796
  ... 797
  ... 798
  ... 799
  ... 800
  ... 801
  ... 802
  ... 803
  ... 804
  ... 805
  ... 806
  ... 807
  ... 808
  ... 809
  ... 810
  ... 811
  ... 812
  ... 813
  ... 814
  ... 815
  ... 816
  ... 817
  ... 818
  ... 819
  ... 820
  ... 821
  ... 822
  ... 823
  ... 824
  ... 825
  ... 826
  ... 827
  ... 828
  ... 829
  ... 830
  ... 831
  ... 832
  ... 833
  ... 834
  ... 835
  ... 836
  ... 837
  ... 838
  ... 839
  ... 840
  ... 841
  ... 842
  ... 843
  ... 844
  ... 845
  ... 846
  ... 847
  ... 848
  ... 849
  ... 850
  ... 851
  ... 852
  ... 853
  ... 854
  ... 855
  ... 856
  ... 857
  ... 858
  ... 859
  ... 860
  ... 861
  ... 862
  ... 863
  ... 864
  ... 865
  ... 866
  ... 867
  ... 868
  ... 869
  ... 870
  ... 871
  ... 872
  ... 873
  ... 874
  ... 875
  ... 876
  ... 877
  ... 878
  ... 879
  ... 880
  ... 881
  ... 882
  ... 883
  ... 884
  ... 885
  ... 886
  ... 887
  ... 888
  ... 889
  ... 890
  ... 891
  ... 892
  ... 893
  ... 894
  ... 895
  ... 896
  ... 897
  ... 898
  ... 899
  ... 900
  ... 901
  ... 902
  ... 903
  ... 904
  ... 905
  ... 906
  ... 907
  ... 908
  ... 909
  ... 910
  ... 911
  ... 912
  ... 913
  ... 914
  ... 915
  ... 916
  ... 917
  ... 918
  ... 919
  ... 920
  ... 921
  ... 922
  ... 923
  ... 924
  ... 925
  ... 926
  ... 927
  ... 928
  ... 929
  ... 930
  ... 931
  ... 932
  ... 933
  ... 934
  ... 935
  ... 936
  ... 937
  ... 938
  ... 939
  ... 940
  ... 941
  ... 942
  ... 943
  ... 944
  ... 945
  ... 946
  ... 947
  ... 948
  ... 949
  ... 950
  ... 951
  ... 952
  ... 953
  ... 954
  ... 955
  ... 956
  ... 957
  ... 958
  ... 959
  ... 960
  ... 961
  ... 962
  ... 963
  ... 964
  ... 965
  ... 966
  ... 967
  ... 968
  ... 969
  ... 970
  ... 971
  ... 972
  ... 973
  ... 974
  ... 975
  ... 976
  ... 977
  ... 978
  ... 979
  ... 980
  ... 981
  ... 982
  ... 983
  ... 984
  ... 985
  ... 986
  ... 987
  ... 988
  ... 989
  ... 990
  ... 991
  ... 992
  ... 993
  ... 994
  ... 995
  ... 996
  ... 997
  ... 998
  ... 999
  ... 1000
```

```
const colorRatio = Math.round(ratio * 100);

let color = tinycolor({
  h: 100 - colorRatio * 0.8,
  s: 100,
  l: 50,
  a: colorRatio / 100
})
.toRgbString();

el.parentElement.style.backgroundColor = color;
el.parentElement.style.borderColor = color;
el.innerHTML = `${Math.floor(val)}';
}

Object.keys(data.values).forEach(num => {
  let d = String.fromCharCode(parseInt(num, 10) + 64);
  const vbox = this.$refs[`detector${d}`];
  if(!vbox || vbox.length === 0 || data.values[num].length === 0) return;
  const el = vbox[0];
  const sourceValues = Object.values(data.values[num]);
  const avg = sourceValues.reduce((acc, source) => source.ac + acc, 0)
    / sourceValues.length;

  updateElement(el, avg);
});

const list = Object.values(data.values);
this.layout.forEach(({ address, type }) => {
  if(type === 'detector') return;
  const vbox = this.$refs[`source${address}`];
  if(vbox.length === 0) return;
  const el = vbox[0];

  const values = list
    .filter(det => det[address])
    .map(det => det[address]);

  const avg = values
    .reduce((acc, curr) => acc + curr.ac, 0) / values.length;

  updateElement(el, avg);
});

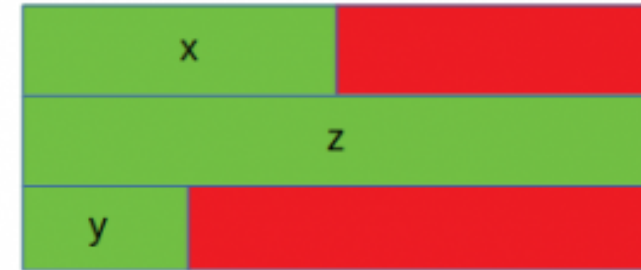
genGraph(layout) {
  const { depth } = this;
  const distance = (a, b) => Math.pow((a.x - b.x), 2) + Math.pow((a.y - b.y), 2)
  return layout
    .filter(x => x.type === 'detector')
    .map(x => {
      const distances = layout
        .reduce((acc, y) => {
          if(y.i === x.i || x.type === y.type) return acc;
          const d = Math.floor(distance(x, y));
          if(!acc[d]) acc[d] = [];
          acc[d].push(y);
        }, {});
      return distances;
    });
}
```


Technology requirements

Building the solution

- Communication with DMC through raw TCP/IP
- Ability to receive **binary messages** and convert them to **C++ structures** with specified variable offset
- Modern, **fast**, scalable and maintainable framework

```
0000 00 14 6c 51 db 04 00 40 2b 65 41 24 08 00 45 00
0010 04 33 b0 56 40 00 80 06 5c 08 c0 a8 03 03 44 8e
0020 e2 2c 0f 40 00 50 0e 9e 18 87 dc 5b 6e 63 50 18
0030 ff ff 43 9f 00 00 47 45 54 20 2f 20 48 54 54 50
0040 2f 31 2e 31 0d 0a 48 6f 73 74 3a 20 77 77 77 2e
0050 79 61 68 6f 6f 2e 63 6f 6d 0d 0a 55 73 65 72 2d
0060 41 67 65 6e 74 3a 20 4d 6f 7a 69 6c 6c 61 2f 35
0070 2e 30 20 28 57 69 6e 64 6f 77 73 3b 20 55 3b 20
0080 57 69 6e 64 6f 77 73 20 4e 54 20 35 2e 31 3b 20
0090 65 6e 2d 55 53 3b 20 72 76 3a 31 2e 38 2e 30 2e
00a0 33 29 20 47 65 63 6b 6f 2f 32 30 30 36 30 34 32
```



Technology selection

Building the solution



Vue.js

Front-end framework



SignalR

Communication framework



.NET Core

Back-end framework

+



Electron



VS Code



ESLint



Technology selection – .NET Core

Building the solution



.NET Core

Back-end framework

Works with:



Free. Cross-platform. Open source.



Console



Gaming



Web



Machine Learning



Mobile



Internet of Things



Desktop

Technology selection – SignalR

Building the solution



SignalR

Communication framework



Fast and scalable framework for real-time communication.
Already part of .NET Core framework

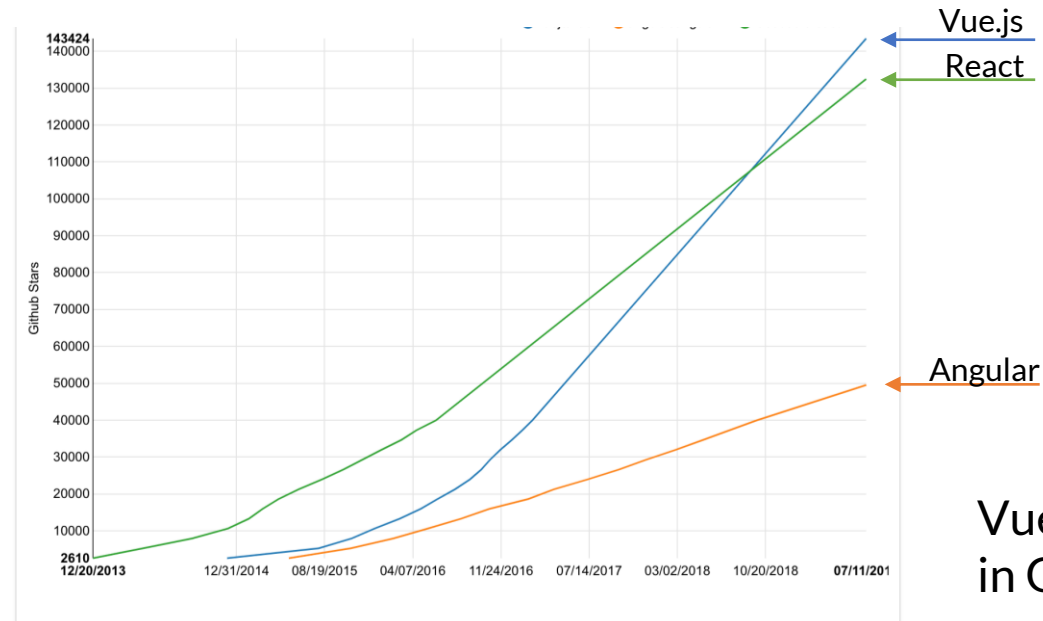
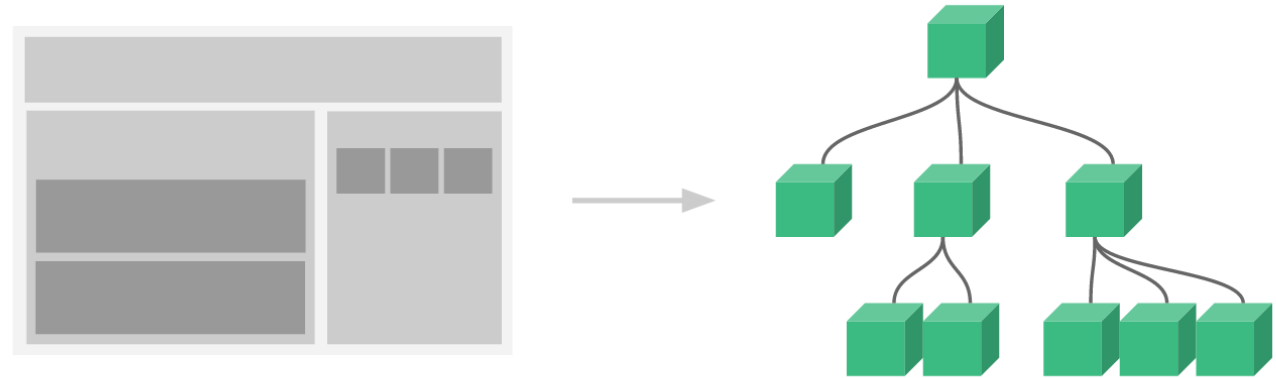
Technology selection – Vue.js

Building the solution



Vue.js

Front-end framework



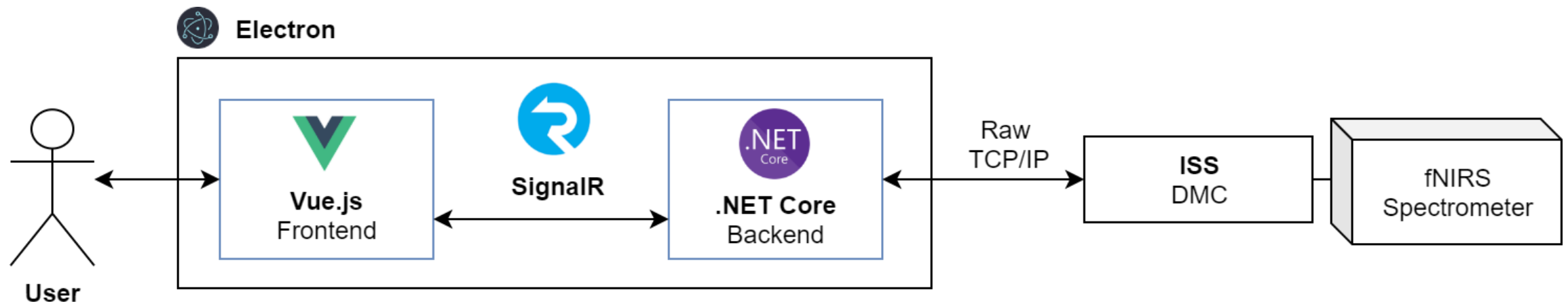
Vue.js popularity
in GitHub stars

Technology selection - Electron

Building the solution



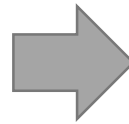
Build cross platform desktop apps with **JavaScript**, **HTML**, and **CSS**



Technology selection - Electron

Building the solution

Pending request from community



My custom template for Electron + .NET Core + Vue.js available on GitHub

Add electron.net template for vue.js and angular #265

Open miladbonakdar opened this issue on 31 May 2019 · 2 comments New issue

miladbonakdar commented on 31 May 2019

I saw on your documents that you shared a template for React as a UI framework. Can you add a simple template for getting started with Vue.js or Angular? I think it would be a good idea to add those two. I am trying to use vue.js pure javascript for my app but I cannot use Vue templates because all of them using Vue cli to run. I will appreciate if you at least give me a hint on how to do that.

miladbonakdar added the **Feature** label on 31 May 2019

netpoetica commented on 3 Jun 2019

Can you describe the problem in a bit more detail? What is your level of expertise in Vue and Angular? Because I think I could probably help you get started, but most likely, project templates for Vue and Angular would best be left in userland with Vue/Angular experts - maybe we could help you get the Electron.NET ecosystem synchronized with your Vue/Angular app and you could be the template creator :)

I am not very familiar with Vue CLI at all (seems crazy to me that a CLI tool would be used for a front-end app framework? Except for maybe the build process). If you can put together a sample repo and some step by step of what commands you are running, maybe we could see the disconnect and get it sorted

johndab commented on 20 Oct 2019

I've just created a template with Electron.NET and Vue by using .NET Core VueCliMiddleware
Here: <https://github.com/johndab/electron-net-vue>

You can have standalone Vue project running independently and just proxy requests from .net to vue in development.
For production it just generates html and js static files.

The only issues I have is that `env.IsDevelopment()` doesn't work in `Startup.cs` with Electron.NET because the project is being published by the Electron.CLI.
And `AfterTargets="ComputeFilesToPublish"` in `csproj` is executed every time I start the development -

Assignees
No one assigned

Labels
Feature

Projects
None yet

Milestone
No milestone

Notifications Customize
[Unsubscribe](#)
You're receiving notifications because you commented.

3 participants

johndab / **electron.net-vue** Unwatch 1 Star 1 Fork 0

[Code](#) [Issues 0](#) [Pull requests 0](#) [Actions](#) [Projects 0](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

Electron.NET with Vue project template Edit

[Manage topics](#)

5 commits 1 branch 0 packages 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find file Clone or download

johndab Add target Latest commit cdb9c7 on 20 Oct 2019

Controllers	Cleanup	3 months ago
Properties	Init	3 months ago
front-end	Add target	3 months ago
.gitignore	keep wwwroot	3 months ago
Program.cs	Init	3 months ago
README.md	Add target	3 months ago
Startup.cs	Cleanup	3 months ago
appsettings.Development.json	Init	3 months ago
appsettings.json	Init	3 months ago
electron.manifest.json	Init	3 months ago
electron-net-vue.csproj	Cleanup	3 months ago

README.md

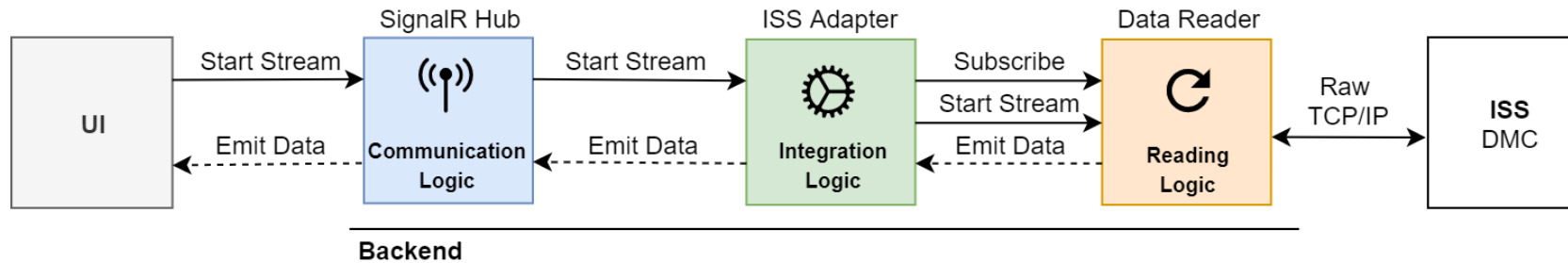
Running demo:

Development

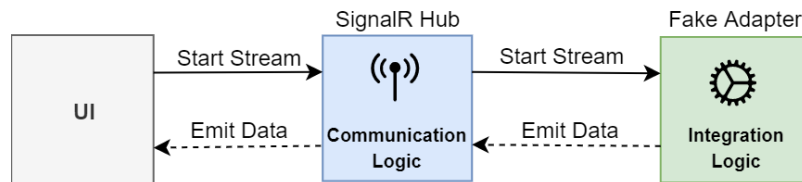
Data Processing

Building the solution

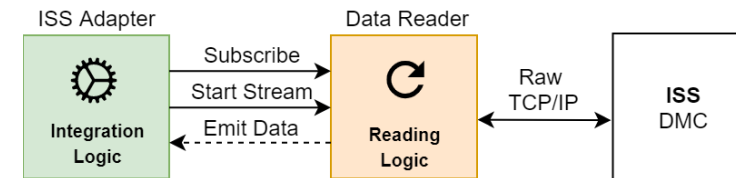
Data stream flow



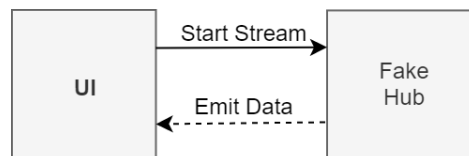
Test Frontend – Backend communication:



Test Backend – DMC communication:



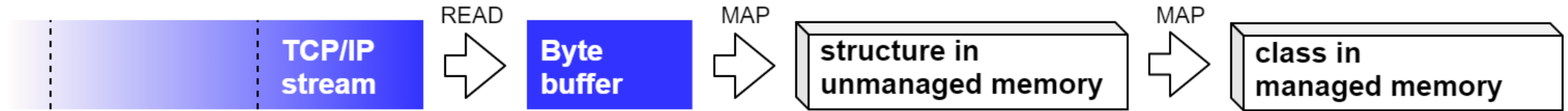
Test Frontend:



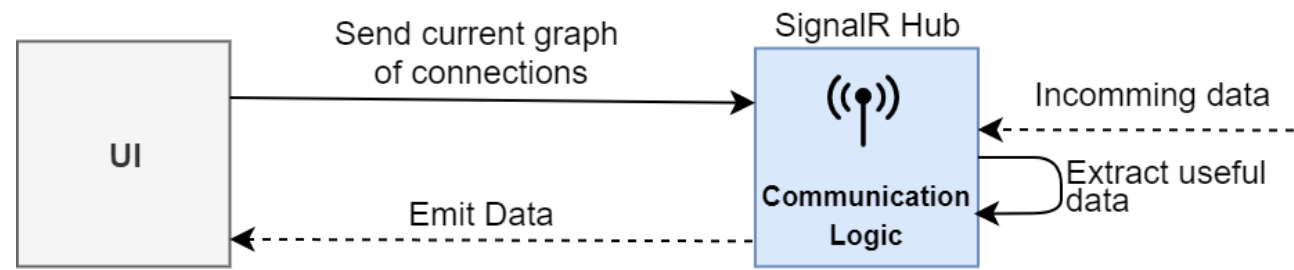
Data Processing

Building the solution

1. Low-level memory management



2. Optimized frontend communication



Data Visualisation

Building the solution

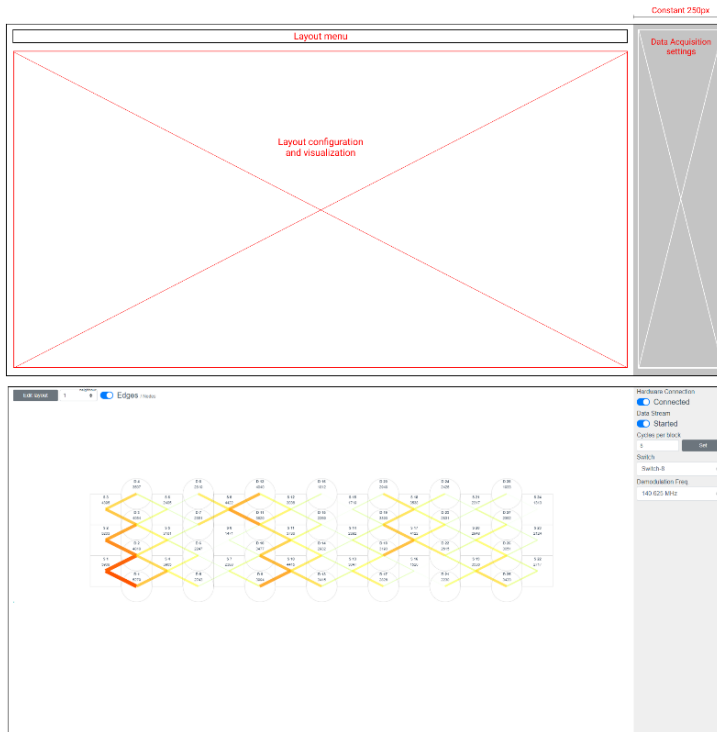
1. Plan the design with wireframes



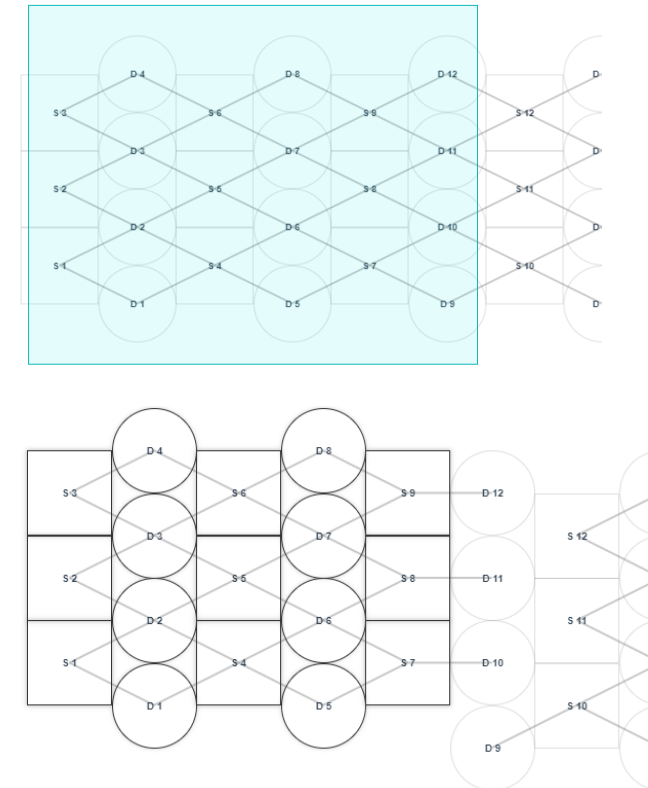
2. Develop components



3. Add user-friendly features



```
▼ <Root>
  ▼ <App>
    ▼ <Home>
      ▼ <Board>
        ▼ <NeighbourSelect>
          <BFormSelect>
          <BFormCheckbox>
        ▼ <Layout>
          ▶ <GridLayout>
          <Connections>
      ▼ <SideBar>
        ▶ <Connection>
        ▶ <Streaming>
        ▶ <CyclesNum>
        ▶ <SwitchMode>
        ▶ <Frequency>
```



Demo

Feature requests

- Auto bias / manual bias settings
- Show 2 frequencies with a switch
- Store acquired data to a file

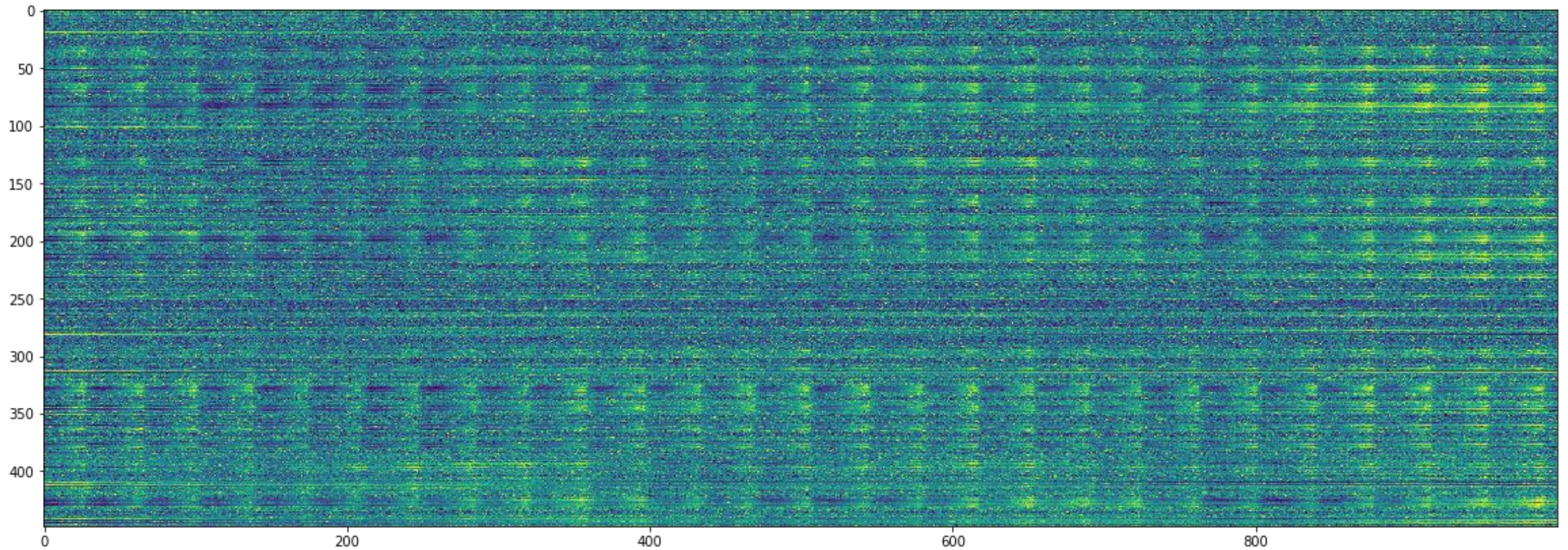
Data Mining

“Data mining is the process of **discovering patterns** in large data sets involving methods at the intersection of **machine learning, statistics, and database systems**”

Wikipedia

Data Mining

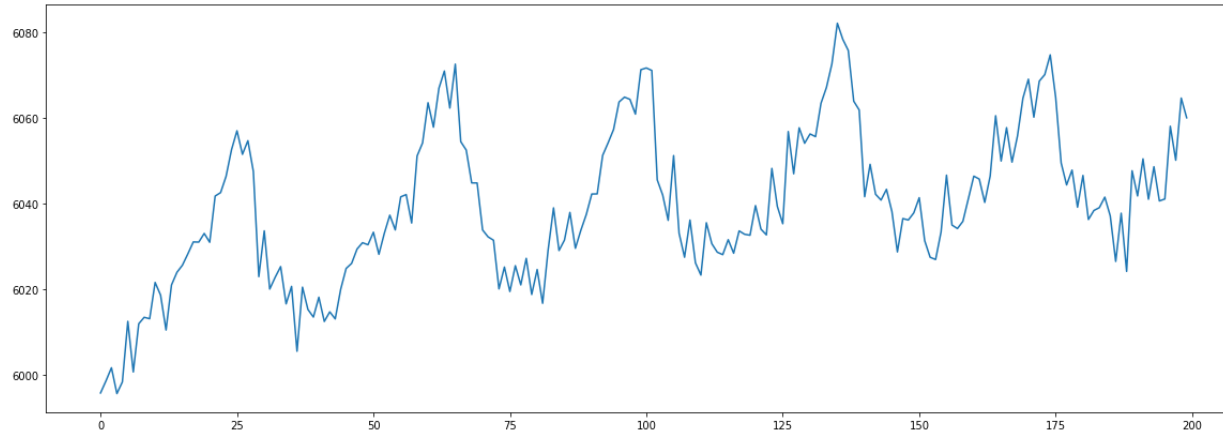
Data exploration



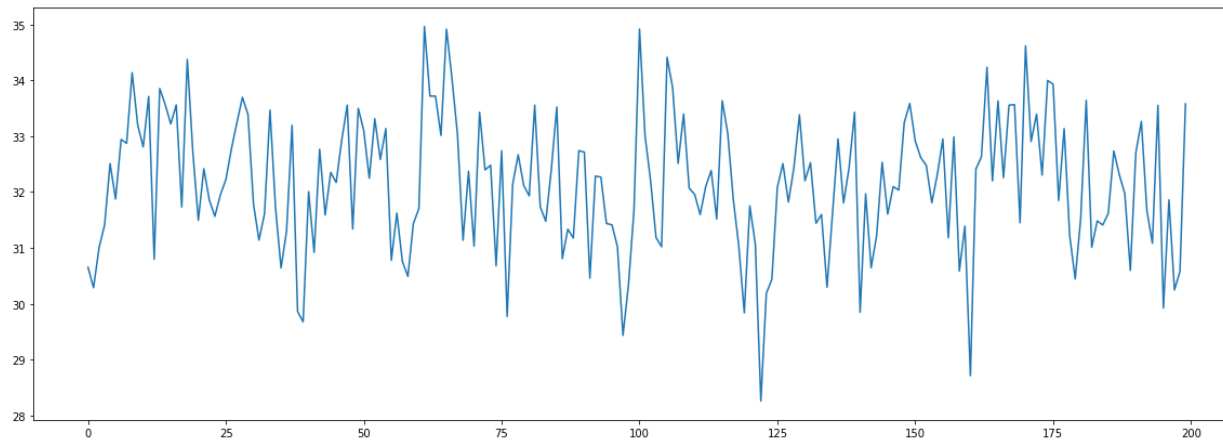
1000 measurements (~25 seconds) across all source-detector channels

Data Mining

Data exploration



Source no. 2 and detector no. 2
for 200 measurements (~5 seconds)

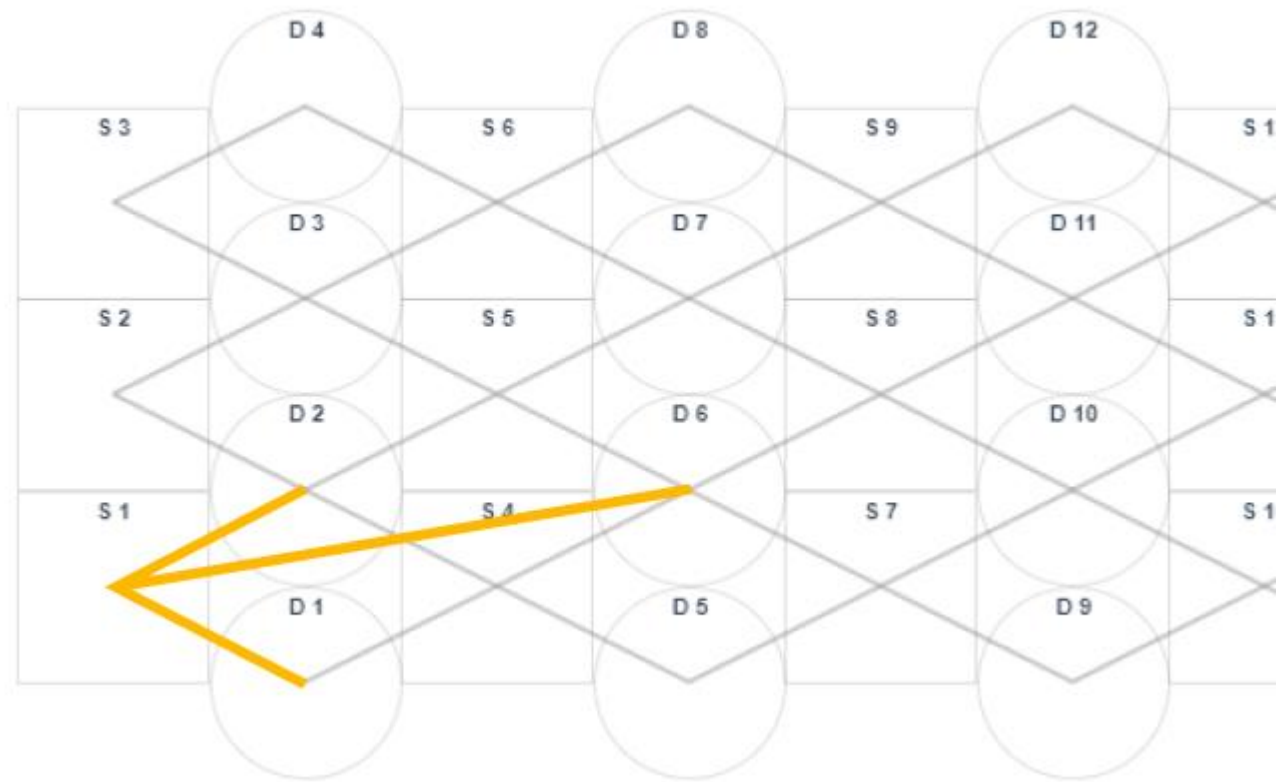
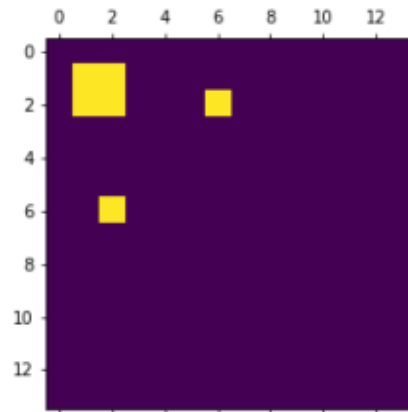
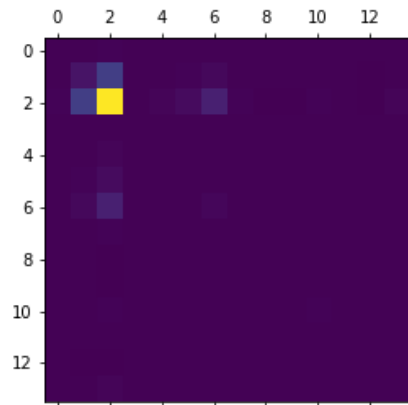


No visible heartbeat oscillation
between source no. 2 and detector
no. 4 for 200 measurements (~5 seconds).

Data Mining

Correlation matrix

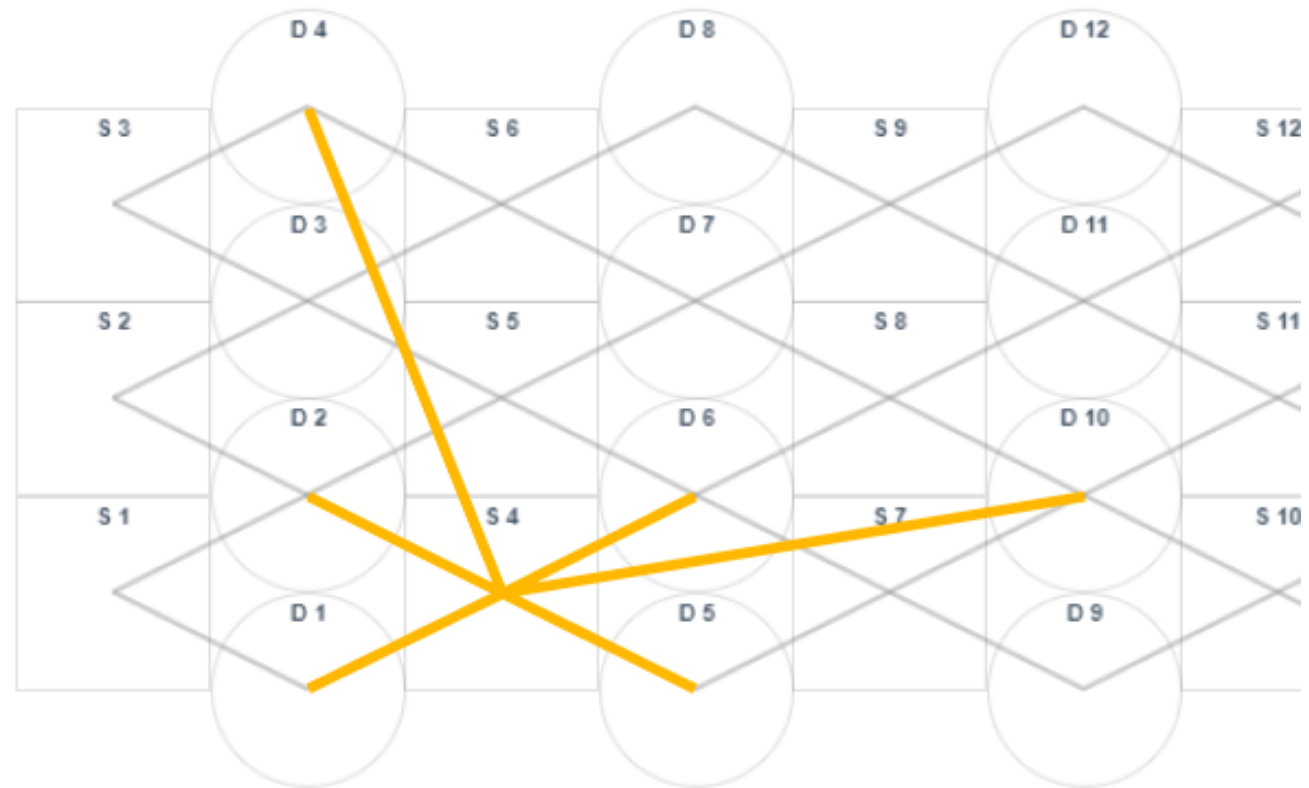
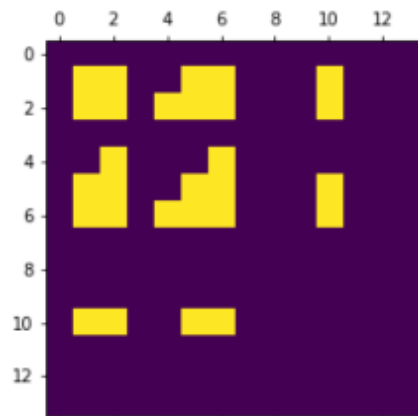
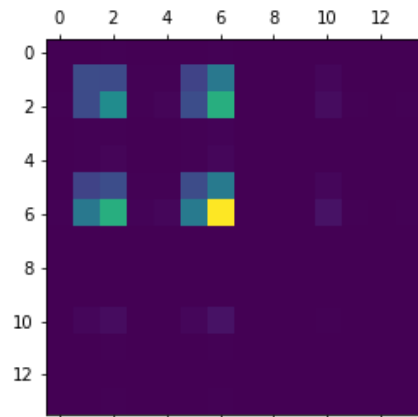
Source 1



Data Mining

Correlation matrix

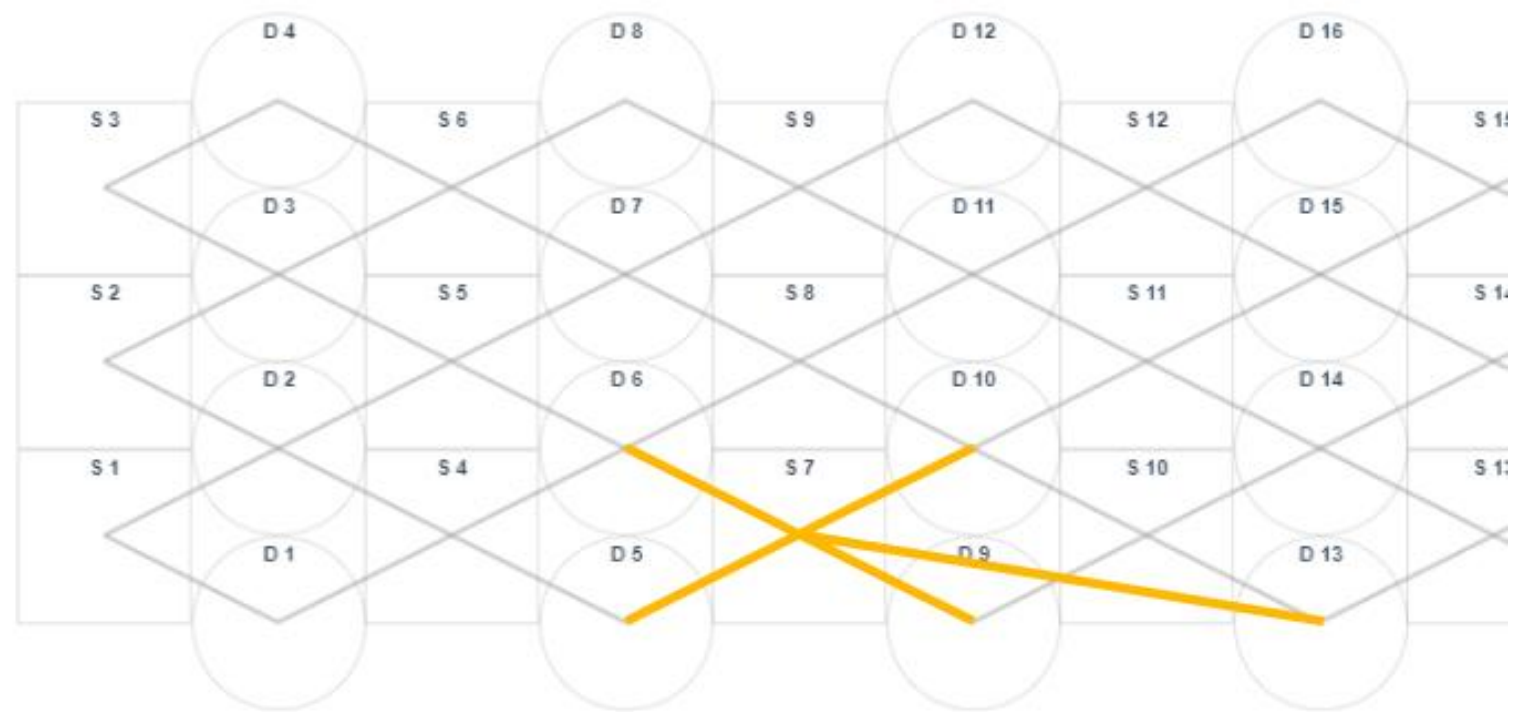
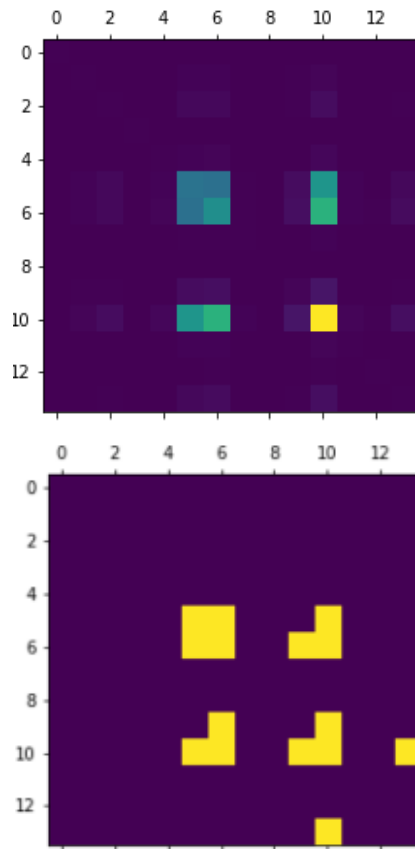
Source 4



Data Mining

Correlation matrix

Source 7



The future

The future

1. Develop a common **standard data format** for functional brain imaging

- easy to create, transmit and store
- real-time processing and storage
- different imaging techniques
- absolute location in the brain

to:

- Use more easily by researchers and practitioners
- Combine different techniques of imaging
- Create re-usable tools for visualisation and analysis

The future

1. Develop a common **standard** for functional brain imaging **data format**

NIfTI

https://nifti.nimh.nih.gov/nifti-1/documentation/hbm_nifti_2004.pdf

DICOM

“the international standard to transmit, store, retrieve, print, process, and display medical imaging information.”

<https://www.dicomstandard.org/>

SNIRF

“Shared Near Infrared File Format”

<https://github.com/fNIRS/snirf>

BIDS

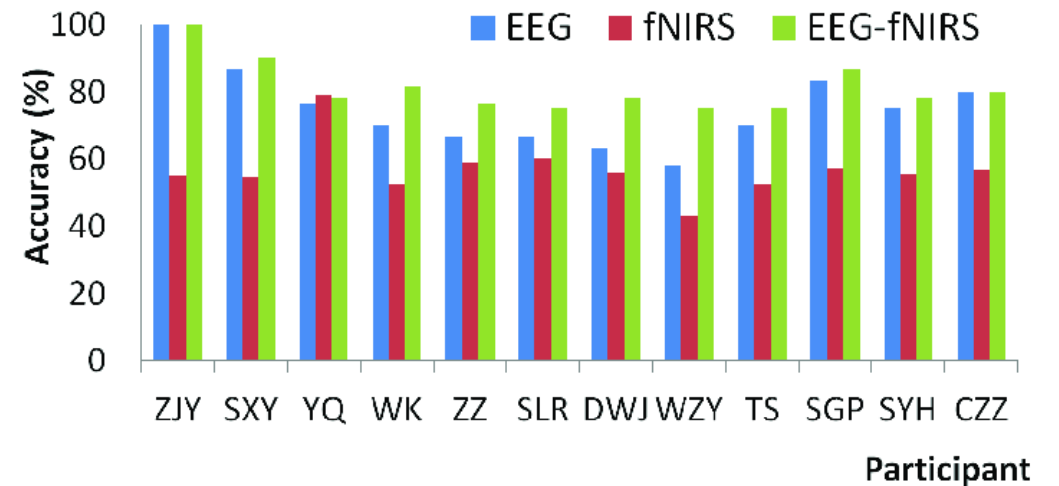
“A format for organizing and describing outputs of neuroimaging experiments”

<https://www.nature.com/articles/sdata201644>

The future

2. Combine EEG and fNIRS

To create cheap, portable and non-invasive, yet powerful diagnostic tool



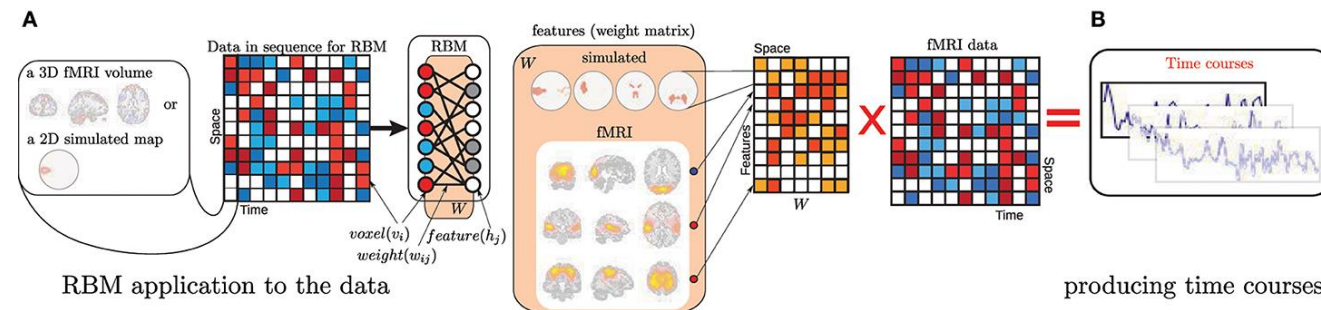
(Ge, et al., 2017)

Combined EEG & fNIRS by Artinis

The future

3. Use Machine Learning

To prepare generalized frameworks for **neural activity pattern recognition**, **injury classification** and **health condition prediction**

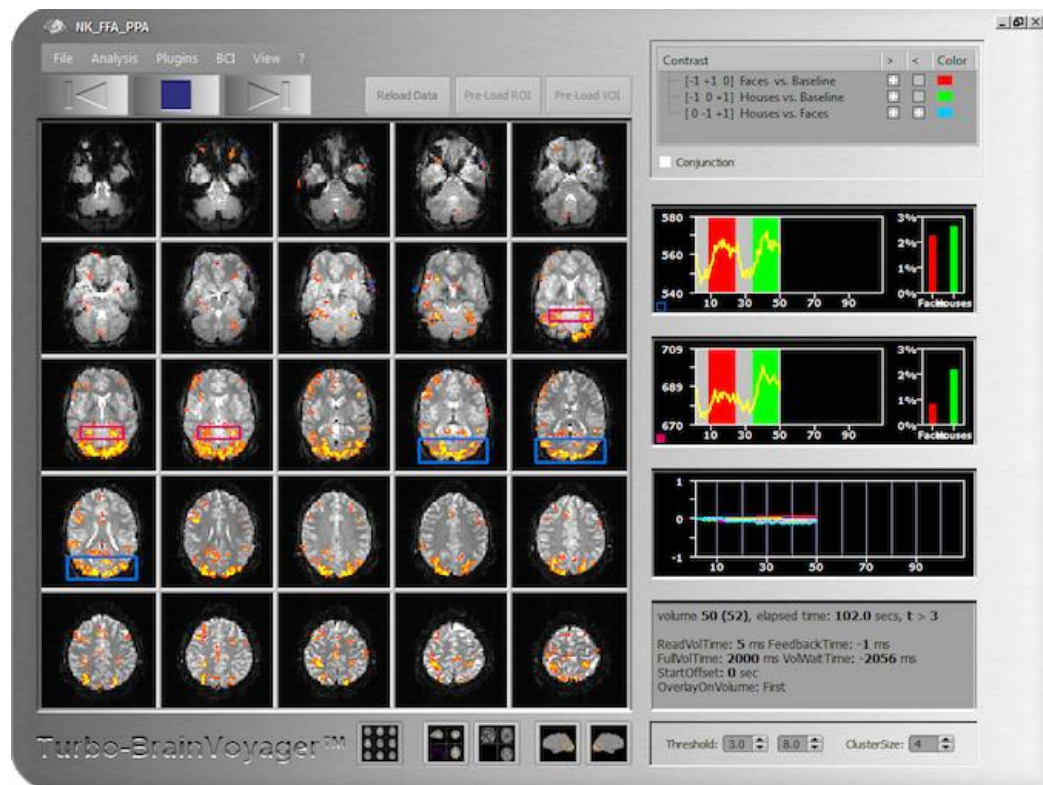


Application of shallow neural network (single hidden layer) on fMRI data to map space-time domain to features
<https://www.frontiersin.org/articles/10.3389/fnins.2014.00229/full>

The future

4. Create **software tools** for doctors and nurses

For everyday tasks



References

Bowman, F.D. (2014) 'Brain Imaging Analysis'
Annu Rev Stat Appl (1), pp. 61-85,
doi:10.1146/annurev-statistics-022513-115611

Dunham, M.H. (2002) *Data Mining: Introductory and Advanced Topics*.
Upper Saddle River, N.J.: Prentice Hall/Pearson Education.

Doulgerakis, M., Eggebrecht, A. and Dehghani, H. (2019)
'Information rich phase content of frequency domain functional Near
Infrared Spectroscopy.', *Neural Imaging and Sensing 2019*, 108650C
doi: 10.1117/12.2505874

GitHub (2019) *Electron Documentation*. Available at:
<https://electronjs.org/docs> (Accessed 07 December 2019).

ISS, Inc. (2019b) Imagent for fNIRS and EROS measurements.
Available at: http://www.iss.com/resources/pdf/technotes/Imagent_fNIRS_EROS_Measurements.pdf
(Accessed 07 December 2019).

References

Kurani, A.S. (2015) 'STRUCTURAL IMAGING'
Available at: <http://www.ajaykurani.com/structural-imaging>
(Accessed 19 January 2020).

Microsoft Corporation (2014) Introduction to SignalR
Available at: <https://docs.microsoft.com/en-us/aspnet/signalr/overview/getting-started/introduction-to-signalr> (Accessed 07 December 2019).

Microsoft Corporation (2019) About .NET Core
Available at: <https://docs.microsoft.com/en-us/dotnet/core/about> (Accessed 07 December 2019).

Vue.js (2019b) Introduction.
Available at: <https://vuejs.org/v2/guide/index.html> (Accessed 07 December 2019).