

COGS138: Neural Data Science

Lecture 4

C. Alex Simpkins, PhD

UCSD Dept. of Cognitive Science, Spring 2023

http://casimpkinsjr.radiantdolphinspress.com/pages/cogs138_sp23

rdrobotics@gmail.com | csimpkinsjr@ucsd.edu

Plan for today

- Announcements
- Review - Last time
- Neural data science data modalities - motion/behavior, eye tracking, gene expression, others
- Tools for Neural data science:
 - PyMO
- Motion capture technology discussion, relevance, issues to be aware of
- Eye trackers
- Gene expression studies introduction
- Asking the right questions in data science

Announcements

- FinAID survey
- A0 - due Friday
- A1 - due a week from release, which will be tonight or tomorrow
- Reading 1 - Released on canvas and in web site password protected area tonight, lecture quiz due next week
- **Waitlist plan!**

Last time

Course links

Website	http://casimpkinsjr.radiantdolphinpress.com/pages/cogs138_sp23	Main face of the course and everything will be linked from here. Lectures, Readings, Handouts, Files, links
GitHub	https://github.com/drsimpkins-teaching	files/data, additional materials & final projects
datahub	https://datahub.ucsd.edu	assignment submission
Piazza	https://piazza.com/ucsd/spring2023/cogs138_sp23_a00/home (course code on canvas home page)	questions, discussion, and regrade requests
Canvas	https://canvas.ucsd.edu/courses/44897	grades, lecture videos
Anonymous Feedback	Will be able to submit via google form	If I ever offend you, use an example you are uncomfortable with, or to provide general feedback. Please remain constructive and polite

What is a program?

- Generally a **program** is a **set of instructions** the programmer defines for a device or entity (usually a computer but not always) to follow
- Regarding computers-> programmer writes a set of instructions (“program”) that tells the computer to perform a set of operations
- When the program is executed, the instructions are carried out
- Does a program have to run on a digital machine? What is a computer? “Multiple realizability”

Why write a program, what does it have to do with neuroscience?

- What do you think? Course discussion...
- Many reasons you may want to write a program
- This can be anything, i.e.:
 - Processing data - behavioral, neural, environmental, etc.
 - Making a robot walk
 - Computer/phone/tablet app for some function

Why python?

- It's free
- Tremendous library support
- Easy interpreted language, quick for prototyping
- Highly optimized computational libraries
- Cross platform/portability
- Strong user community for answering questions/knowledgebase

When python?

- Web app development
- Data science
- Scripting
- Database programming
- Quick prototyping

Why Jupyter Notebooks

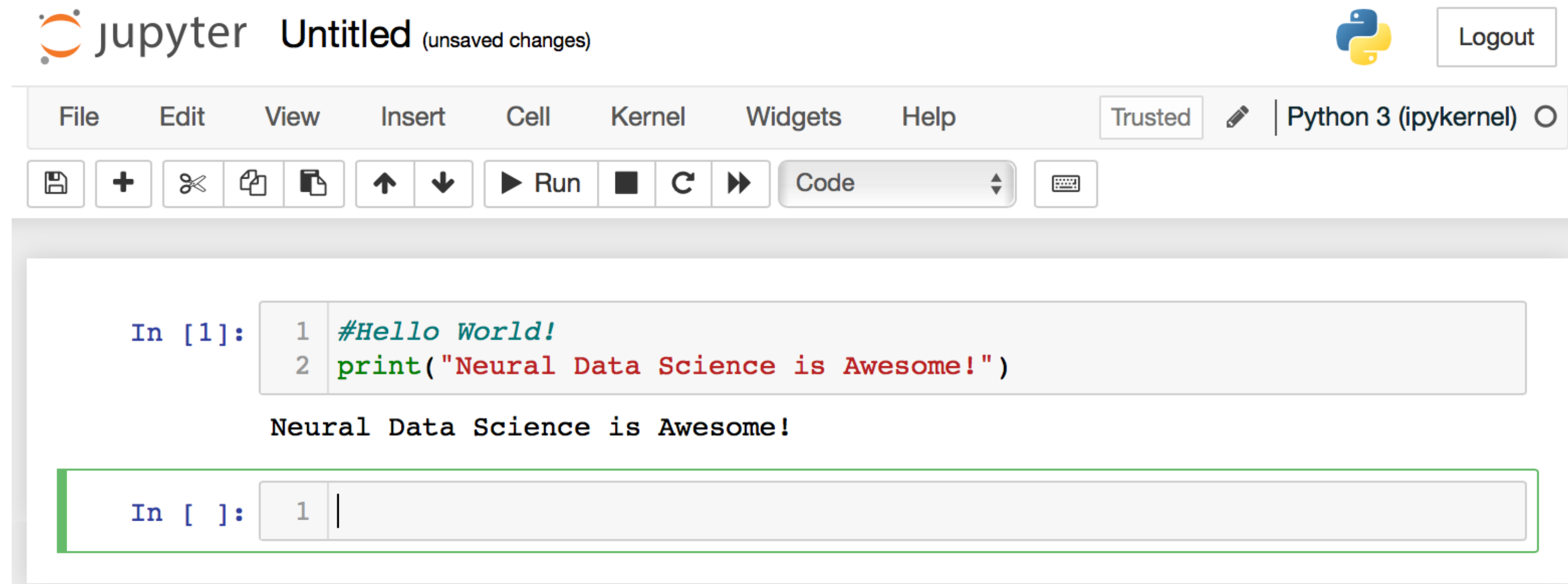
- Mixed media is excellent for data exploration and communication
- Don't have to write a separate program from your notes, results, etc
- Easy to experiment in nonlinear and compartmentalized ways
- We'll discuss the downsides later, but it's not for all cases
 - It can be slow,
 - Version control can be difficult
 - Sometimes debugging is easier other times more difficult

JN use cases

- Prototyping
- Data ingestion
- Exploratory data analysis
- Feature engineering
- Model comparison
- Final model

Jupyter notebooks review

- <https://jupyter.org/>
- Installing [anaconda](#)
- <https://github.com/COGS108/Tutorials>
- <https://github.com/NeuralDataScience/Tutorials>
- Correcting common issues
- Up to students to correct and resubmit so grading can be timely



The screenshot displays the Jupyter Notebook interface. At the top, the title bar reads "jupyter Untitled (unsaved changes)" with the Python logo and a "Logout" button on the right. Below the title bar is a menu bar with options: File, Edit, View, Insert, Cell, Kernel, Widgets, Help. To the right of the menu bar, it shows "Trusted" and "Python 3 (ipykernel)". Below the menu bar is a toolbar with icons for file operations (save, new, copy, paste, undo, redo), a "Run" button, and a "Code" dropdown menu. The main area contains two code cells. The first cell, labeled "In [1]:", contains the following Python code:

```
1 #Hello World!  
2 print("Neural Data Science is Awesome!")
```

The output of this cell is "Neural Data Science is Awesome!". The second cell, labeled "In []:", is currently empty and has a green border around it.

How do you write a program in Jupyter notebooks and python?

- datahub.ucsd.edu
- or your machine with anaconda
- The notebooks we will review are listed below and available in the lectures directory of the github and linked from the website and will be on canvas as well
 - 00-Introduction.ipynb
 - 01-Python.ipynb
 - 02-JupyterNotebooks.ipynb
 - 01_01_python-checkpoint.ipynb

On to today...

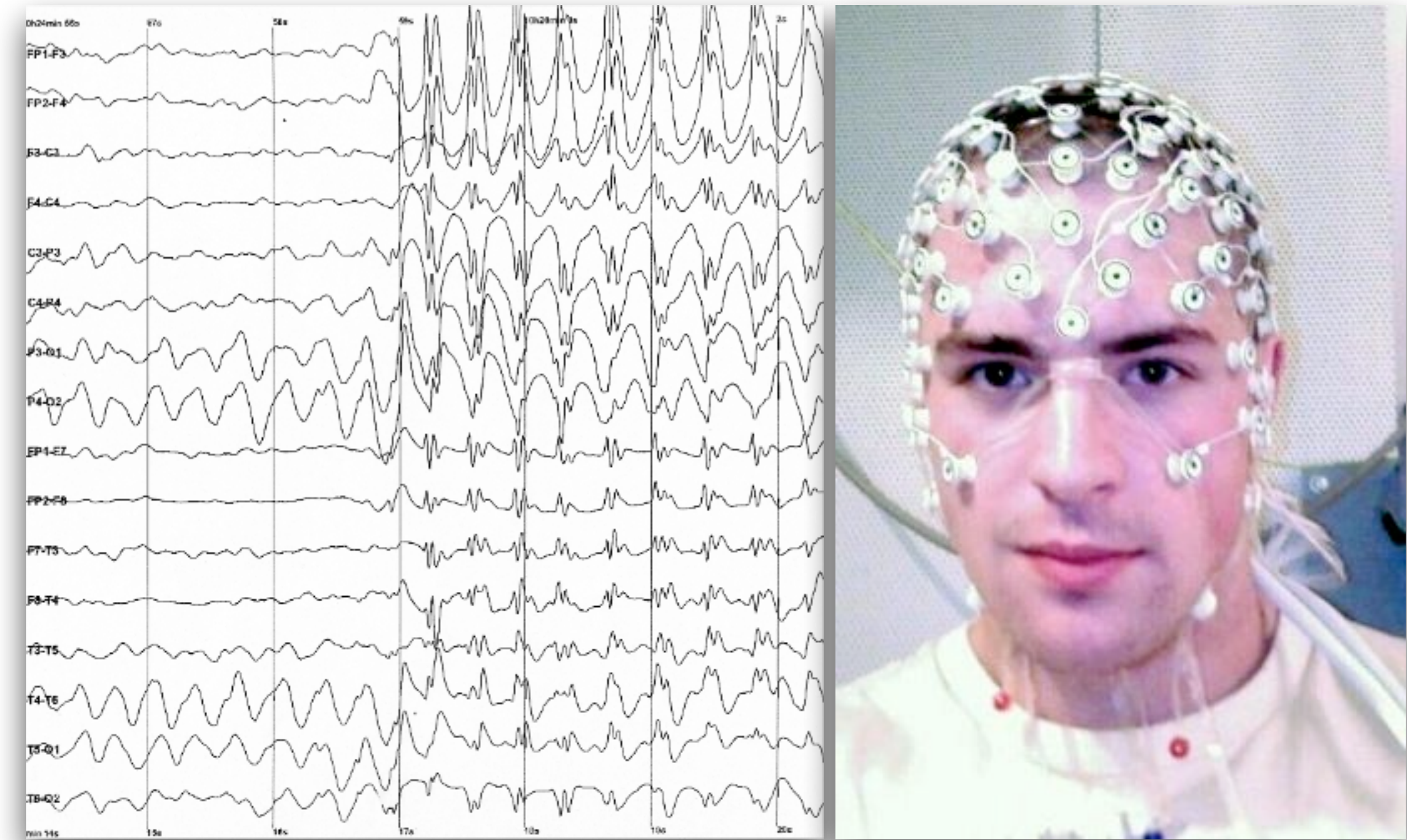
Neural data science toolsets

- There are a variety of toolsets employed in neural data science
- Assist in processing data types, e.g.
 - **EEG and MEG analysis**
 - MNE - <https://mne.tools/stable/index.html>
 - **Linguistics**
 - NLTK- <https://www.nltk.org>
 - **Motion capture data** - kinematic/inverse kinematic and dynamic analysis

Why EEG and MEG analysis?

- **EEG - Electroencephalography**

- Standard location patterns of sensors for recording (20-10, 10-10 systems)
- EEG records electrical activity generated in your brain at the scalp
- Global types of signals such as decision processes, spelling, gross body movement, etc
- Why useful?



(Source: <https://en.wikipedia.org/wiki/Electroencephalography>)

Advantages of EEG

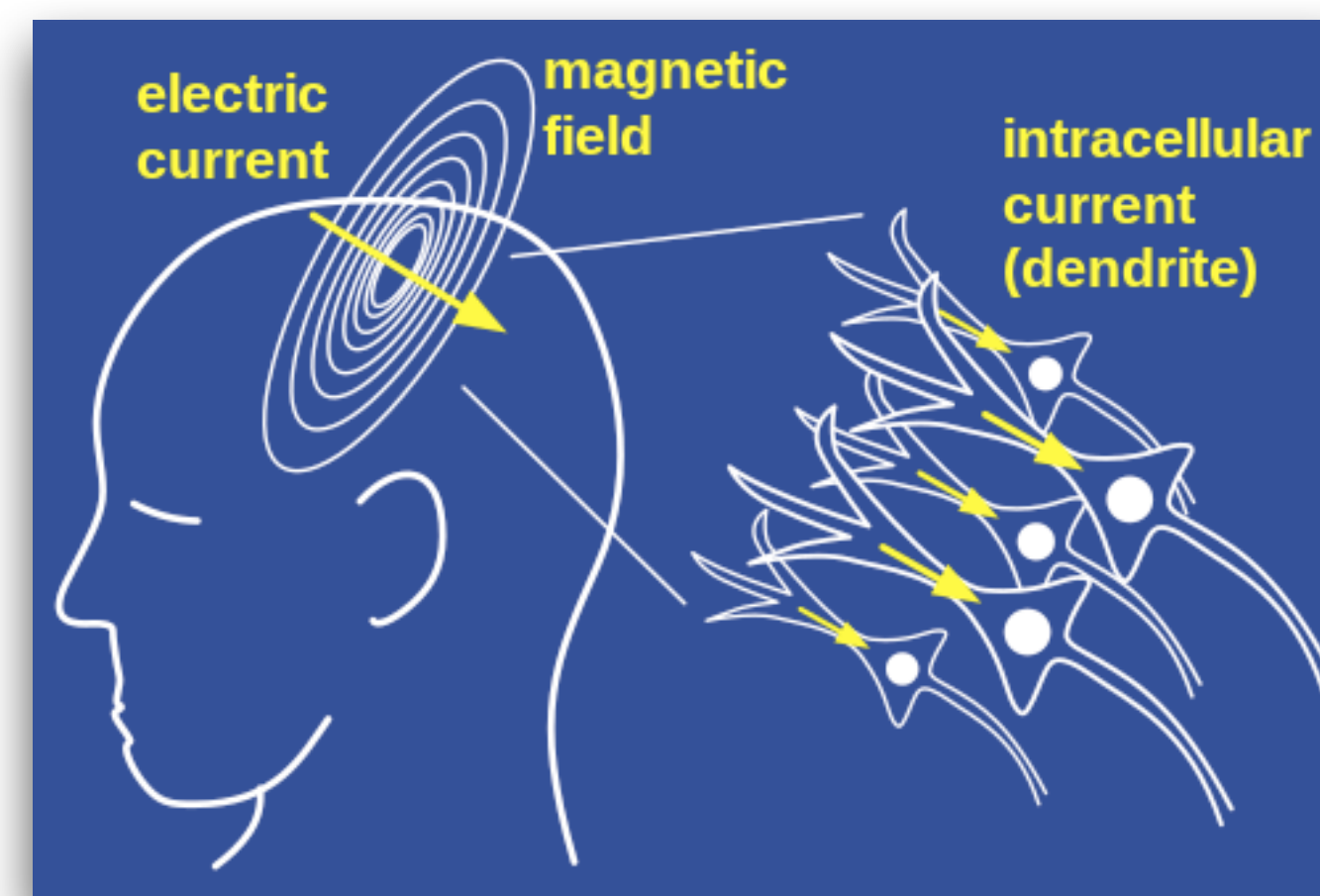
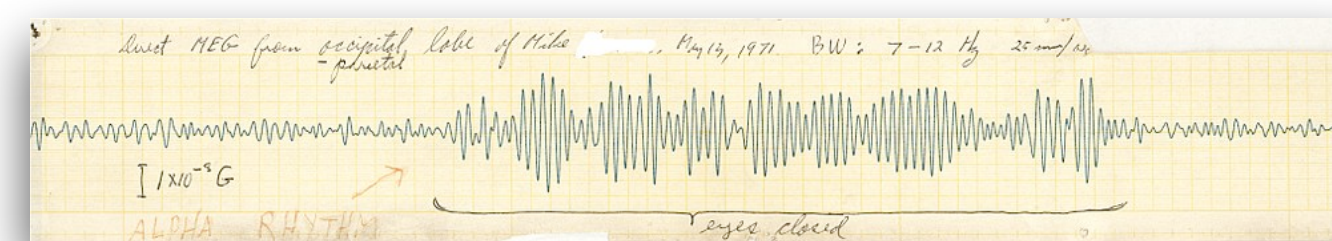
- Low cost
- Small, compact low complexity equipment vs. fMRI, PET, SPECT, MEG, MRS
- Simple data streams to process, signals can be used with minimal pre-processing
- Tolerant of subject movement, unlike many other methods
- Silent - fMRI anyone? Can have metal in nearby space
- No claustrophobia-inducing spaces
- No high intensity magnetic fields, no swallowing radioactive chemicals (PET)
- Better understanding of what signal is being measured than other technologies like fMRI (BOLD)

Disadvantages of EEG

- Low spatial resolution and computational processing required to infer 3d spatial regions that are activated, must make assumptions about internal structure (can be based on scans)
- False localization is possible
- Cannot identify location in brain specific neurotransmitters found
- Slow to connect subjects vs fMRI, MEG, MRS, SPECT
- SNR poor/artifacts (eye movements, heart activity, blinks, facial movements, environment such as 60Hz noise, “Internal noise”)

MEG - Magnetoencephalography

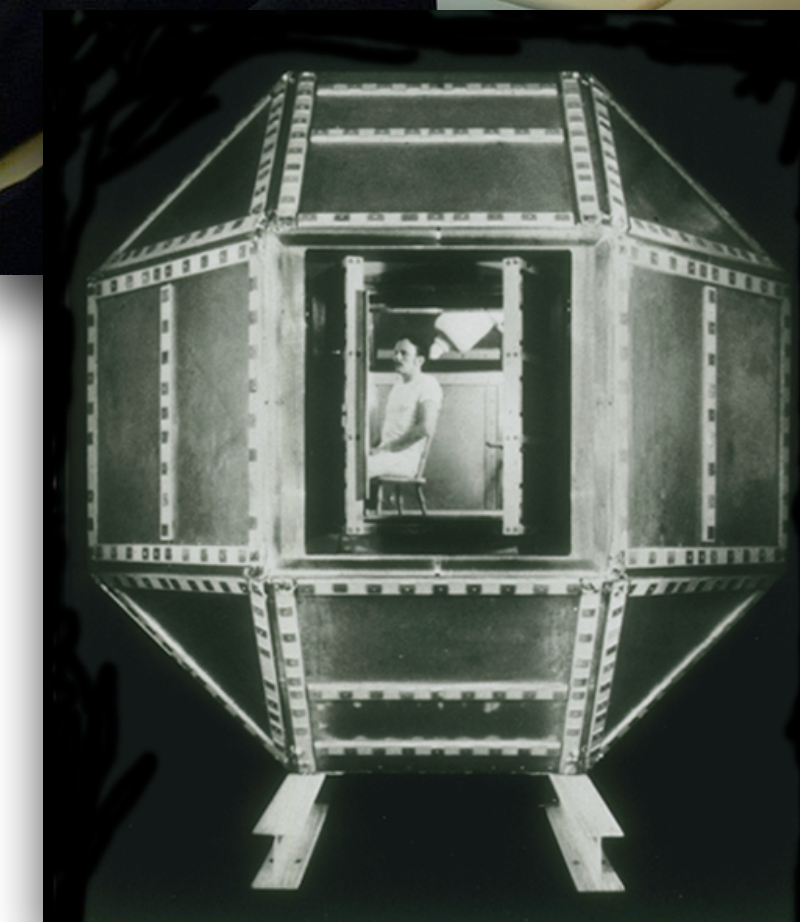
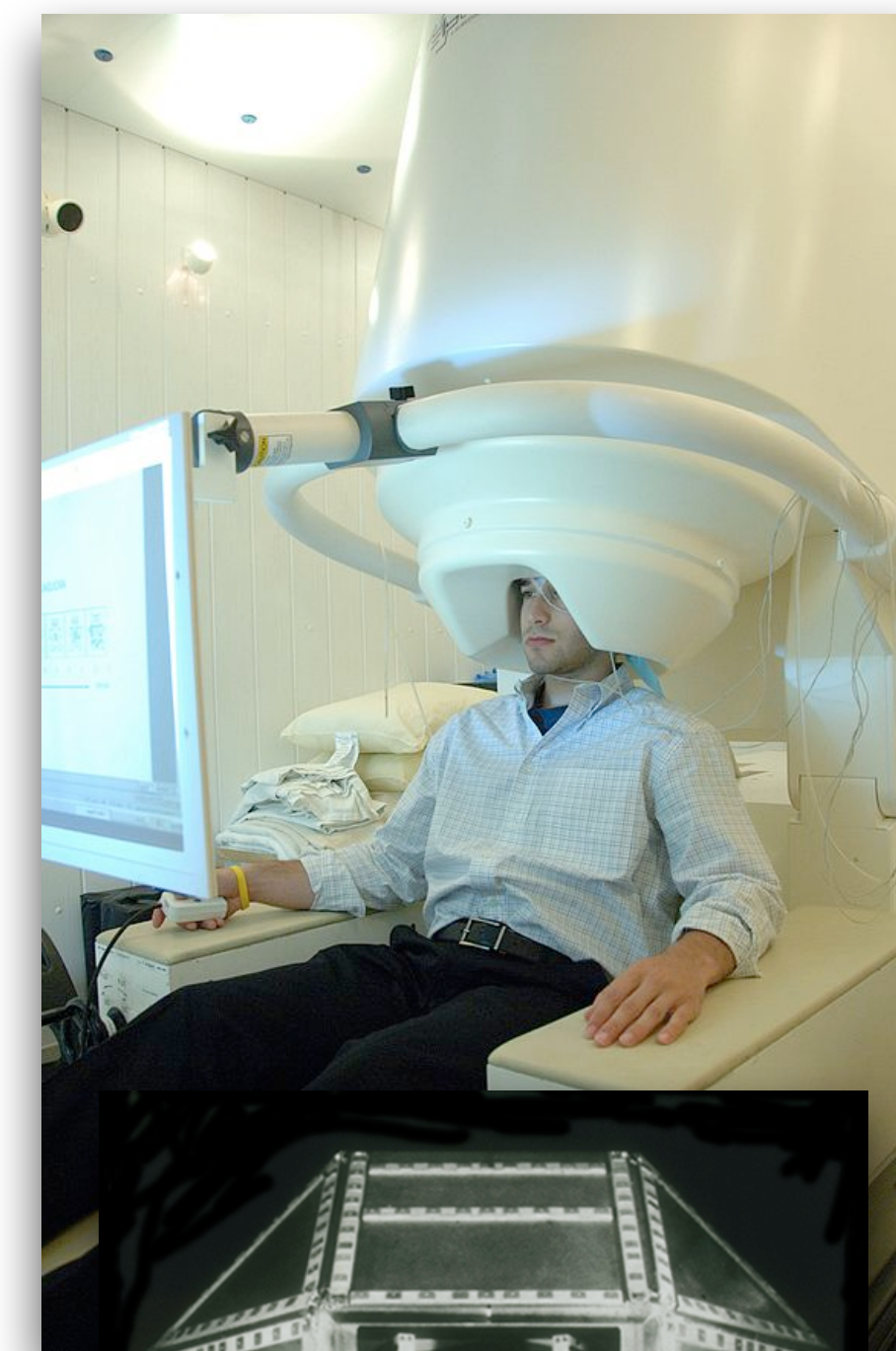
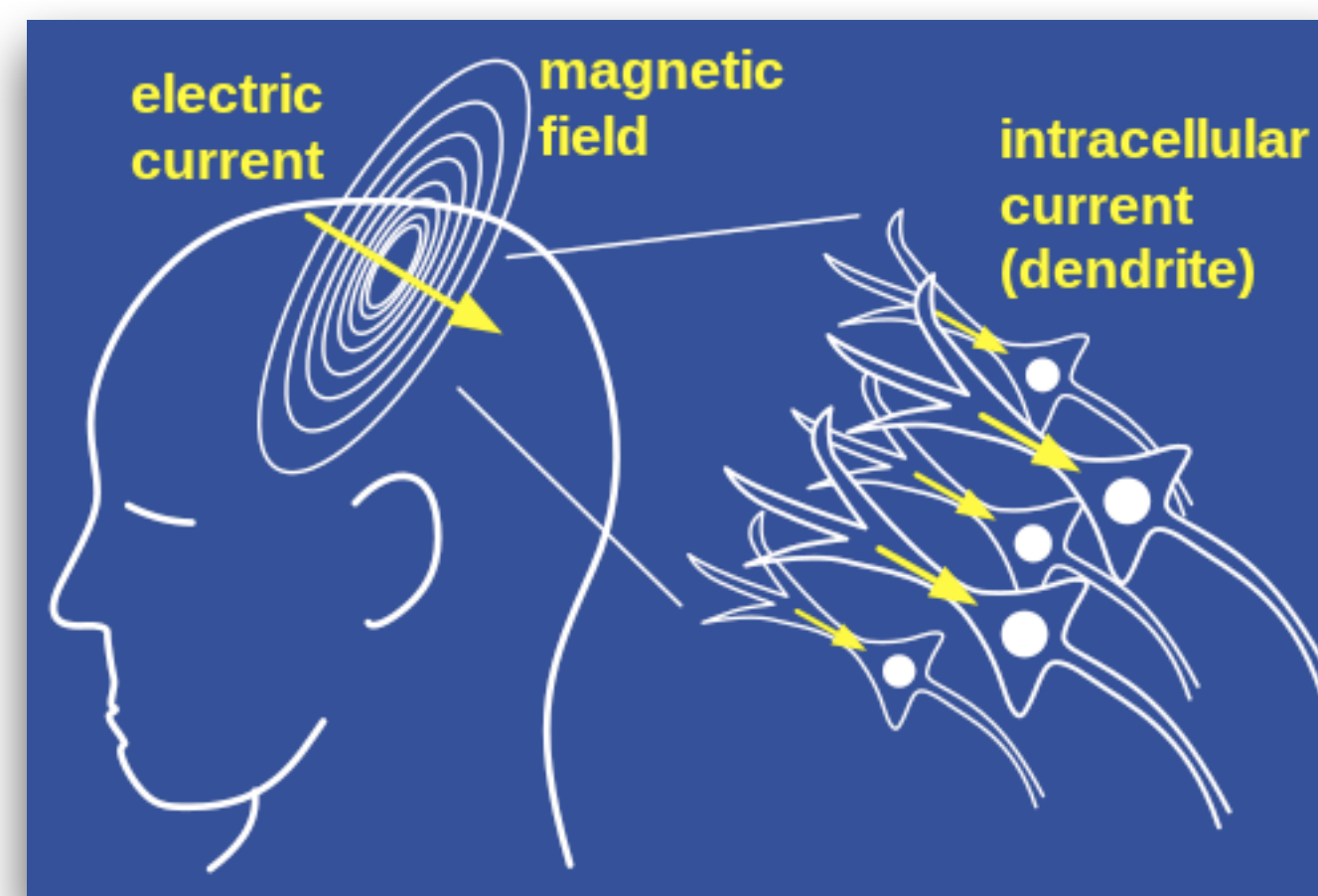
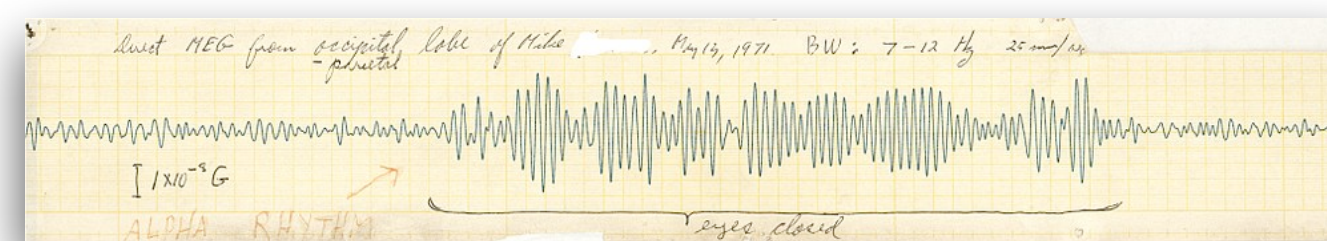
- **MEG** - measurement of the magnetic field generated by electrical activity of neurons
- Mapped onto structural image from MRI
- **Advantages**
 - Provides a higher spatial (mm)/temporal (msec) resolution, no distortion through head
 - Decay relative to dist. is more pronounced than electrical fields thus useful for measuring superficial cortical activity
 - Shows absolute neuronal activity vs. fMRI shows relative activity (fMRI must always be compared to some reference neural activity)
 - Can be recorded for sleeping subjects, unconscious subjects other
 - Safe, no exposure to radiation/emf, noninvasive, easy to use



(Source: <https://en.wikipedia.org/wiki/Magnetoencephalography>)

MEG - Magnetoencephalography

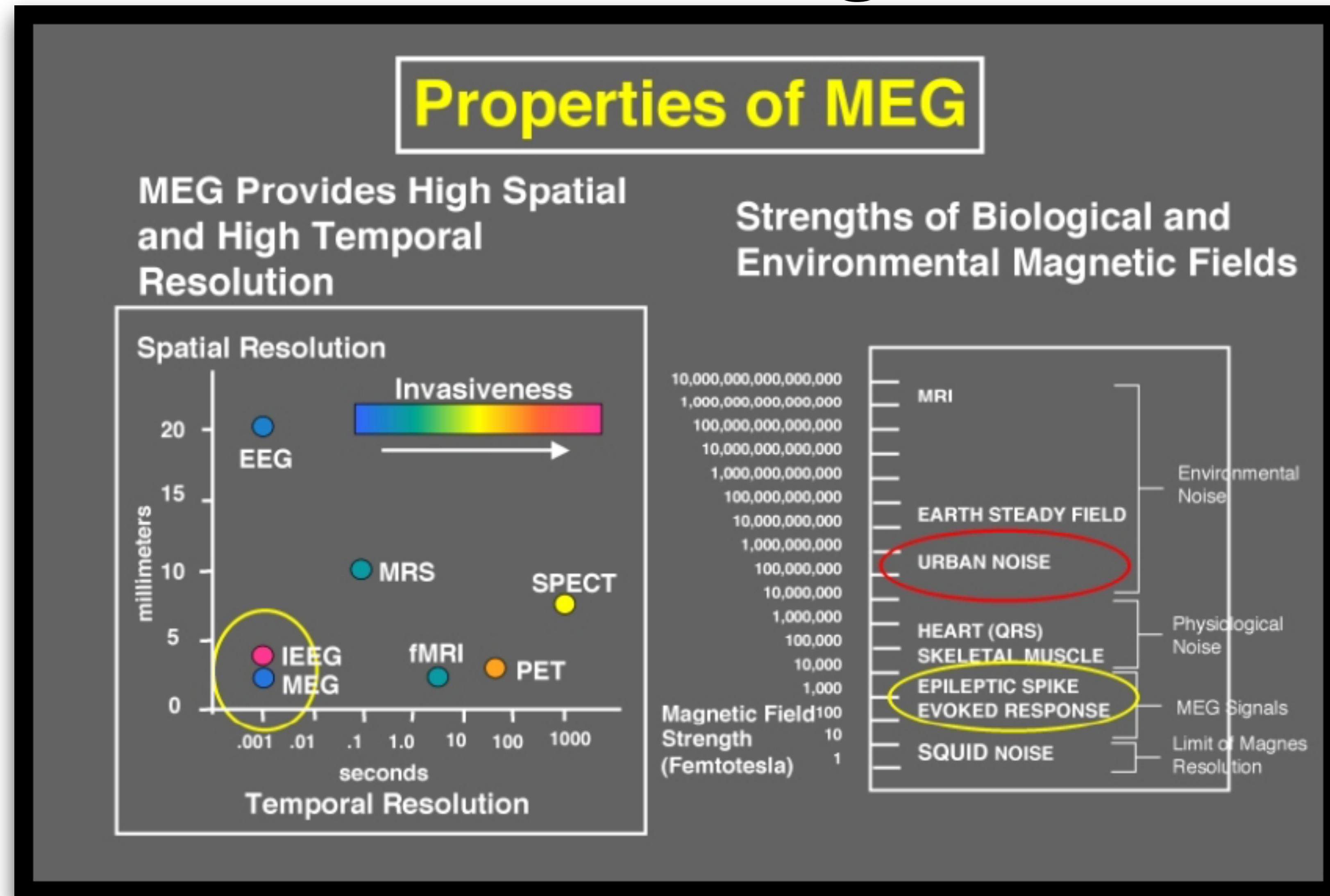
- Disadvantages
 - Patients need to be fairly still
 - Pacemaker or VNS may not allow those patients
 - Possibly non-unique solution to localization problem
 - Sensitive instrumentation needed, subject to environmental noise



Properties and challenges

Problem of biomagnetism:

- The brain's magnetic field, measuring at 10 femtotesla (fT) for cortical activity and 103 fT for the human alpha rhythm
- Ambient magnetic noise in an urban environment, which is on the order of 108 fT or 0.1 μ T
- 50k Neurons for measurement
- Signals must be aligned \rightarrow pyramidal cells (perp. to cortical surface)



Introduction to MNE

- <https://mne.tools/stable/index.html>
- https://mne.tools/stable/auto_tutorials/index.html

Natural Language Processing

- **NLTK** - natural language toolkit (python)
 - <https://www.nltk.org/>
- Easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet
- Libraries for easily performing - classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries
- Documentation and discussion forums

NLTK Simple Examples

- Adding to your notebook or script:

```
import nltk
```

- Checking installation:

```
import nltk
nltk.__path__
```

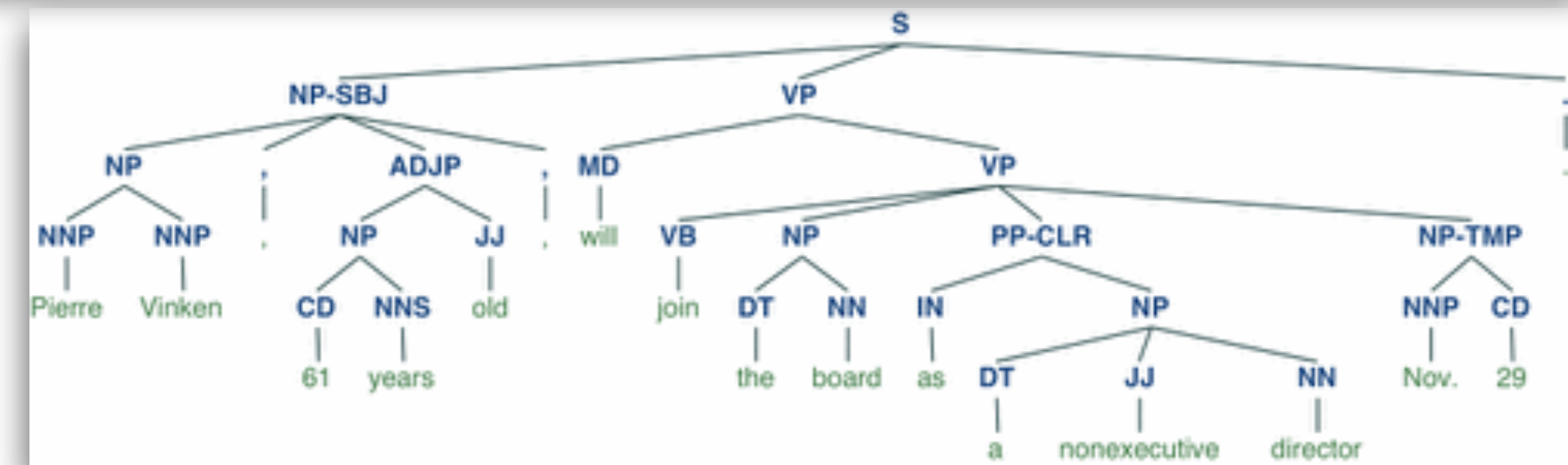
- Tokenize and tag text:

```
nltk.word_tokenize()
nltk.pos_tag()
```

- Display a parse tree:

```
from nltk.corpus import treebank
t = treebank.parsed_sents('wsj_0001.mrg')[0]
t.draw()
```

```
>>> import nltk
>>> sentence = """At eight o'clock on Thursday morning
... Arthur didn't feel very good."""
>>> tokens = nltk.word_tokenize(sentence)
>>> tokens
['At', 'eight', "o'clock", 'on', 'Thursday', 'morning',
'Arthur', 'did', "n't", 'feel', 'very', 'good', '.']
>>> tagged = nltk.pos_tag(tokens)
>>> tagged[0:6]
[('At', 'IN'), ('eight', 'CD'), ("o'clock", 'JJ'), ('on', 'IN'),
('Thursday', 'NNP'), ('morning', 'NN')]
```



NLTK functions


- To page...<https://www.nltk.org/>

Questions we can ask...

1. Does the total number of words change over time?
2. Does uniqueness change over time?
3. Does the diversity or density change?
4. What words are most common?
5. What words are most unique to each year?
6. What sentiment do songs convey most frequently?
7. Has sentiment changed over time?
8. What are the sentiment of the #1 songs?
9. What words contribute to the sentiment of these #1 songs?
10. ...what about bigrams? N-grams?

Sentiment Analysis

Part of the
“NRC”
sentiment
lexicon:



word	sentiment	lexicon
<chr>	<chr>	<chr>
abacus	trust	nrc
abandon	fear	nrc
abandon	negative	nrc
abandon	sadness	nrc
abandoned	anger	nrc
abandoned	fear	nrc
abandoned	negative	nrc
abandoned	sadness	nrc
abandonment	anger	nrc
abandonment	fear	nrc

... with 27,304 more rows

When doing sentiment analysis...

token - a meaningful unit of text

- what you use for analysis
- *tokenization* takes corpus of text and splits it into tokens (words, bigrams, etc.)

stop words - words not helpful for analysis

- extremely common words such as “the”, “of”, “to”
- are typically removed from analysis

When doing sentiment analysis...

stemming - lexicon normalization

- Identifying the root for each token
- Jumping, jumped, jumps, jump all have the same root 'jump'
- Where things get tricky: jumper???

In text analysis, your choices matter:

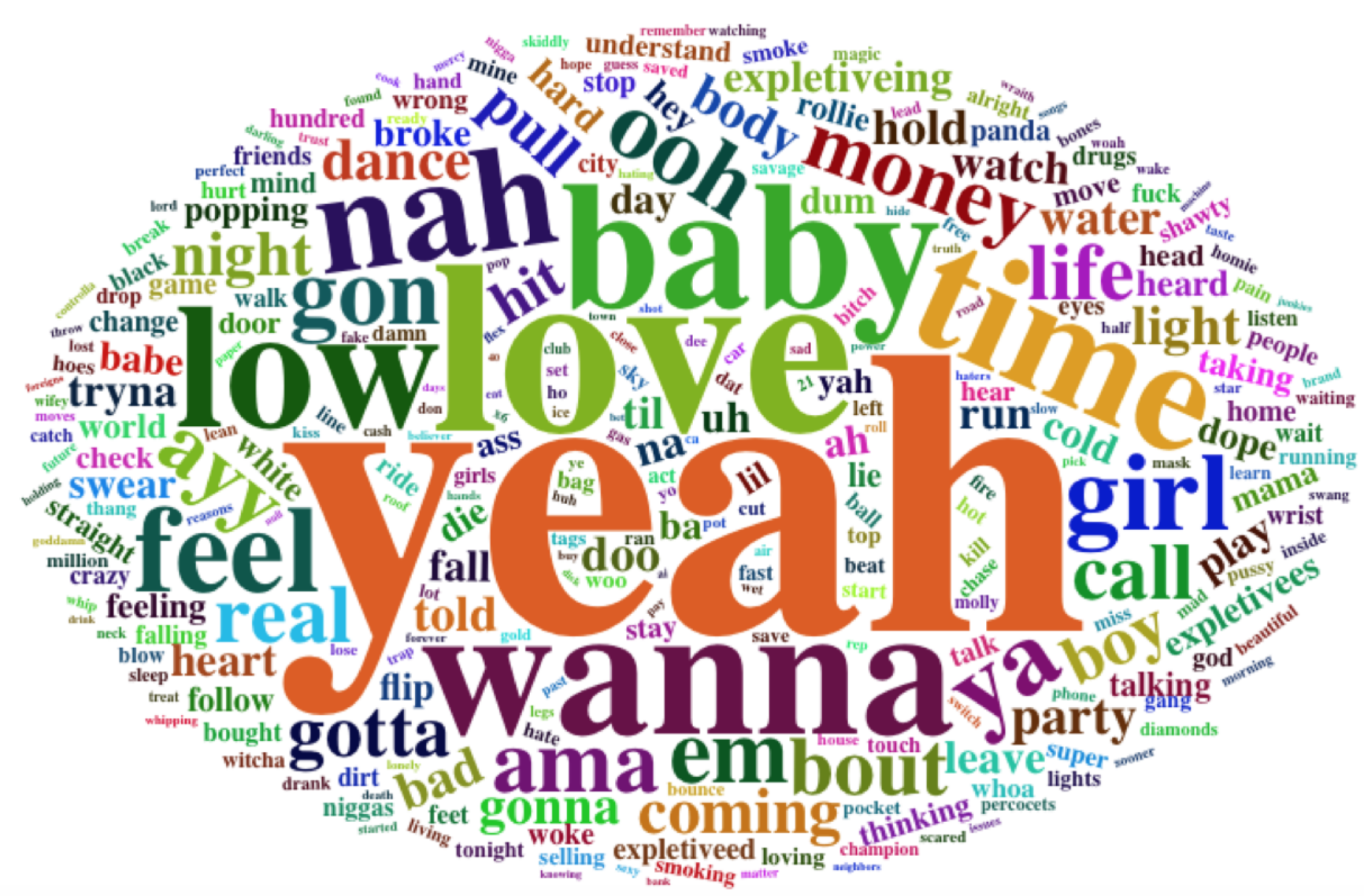
1. How to tokenize?
2. What lexicon to use?
3. Remove stop words? Remove common words?
4. Use stemming?

TF-IDF

Term Frequency - Inverse Document Frequency

What words are the most unique to the lyrics of each year's top hits?

- Goal: to use TF-IDF to *find the important words* for the content of each document by decreasing the weight for commonly used words and increasing the weight for words that are not used very much in a collection or corpus of documents
- Calculating TF-IDF attempts to find the words that are important (i.e., common) in a text, but not *too* common



2017



2018



2019



2020

Term
Frequency
can only tell
us so
much....

TF-IDF:

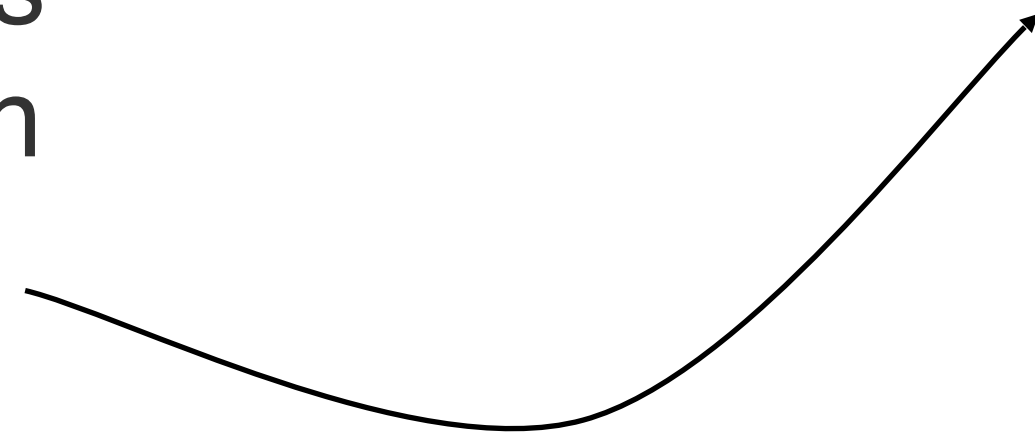
Term Frequency - Inverse Document Frequency

Term Frequency (TF) : how frequently a word occurs in a document

Inverse document frequency (IDF) : intended to measure how important a word is to a document

decreases the weight for commonly used words and increases the weight for words that are not used very much in a collection of documents

$$idf(\text{term}) = \ln \left(\frac{n_{\text{documents}}}{n_{\text{documents containing term}}} \right)$$



TF-IDF:

Term Frequency - Inverse Document Frequency
the frequency of a term adjusted for how rarely it is used

$$w_{x,y} = tf_{x,y} \times \log \left(\frac{N}{df_x} \right)$$

TF-IDF

Term x within document y

$tf_{x,y}$ = frequency of x in y

df_x = number of documents containing x

N = total number of documents

Questions we can ask...

1. Does the total number of words change over time?
2. Does uniqueness change over time?
3. Does the diversity or density change?

EDA

4. What words are most common?
5. What words are most unique to each year?

TF-IDF

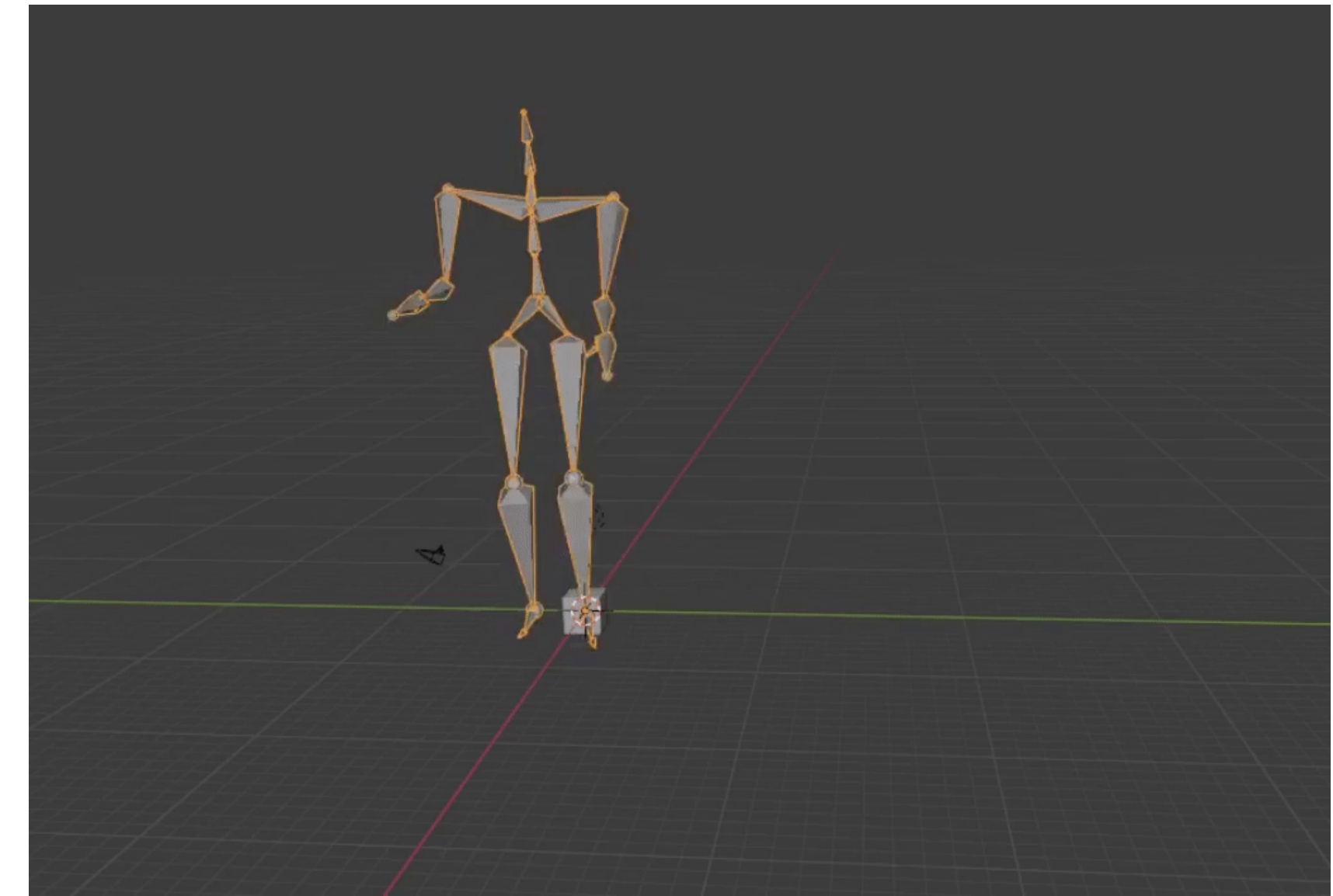
6. What sentiment do songs convey most frequently?
7. Has sentiment changed over time?
8. What are the sentiment of the #1 songs?
9. What words contribute to the sentiment of these #1 songs?
10. ...what about bigrams? N-grams?

Sentiment
Analysis

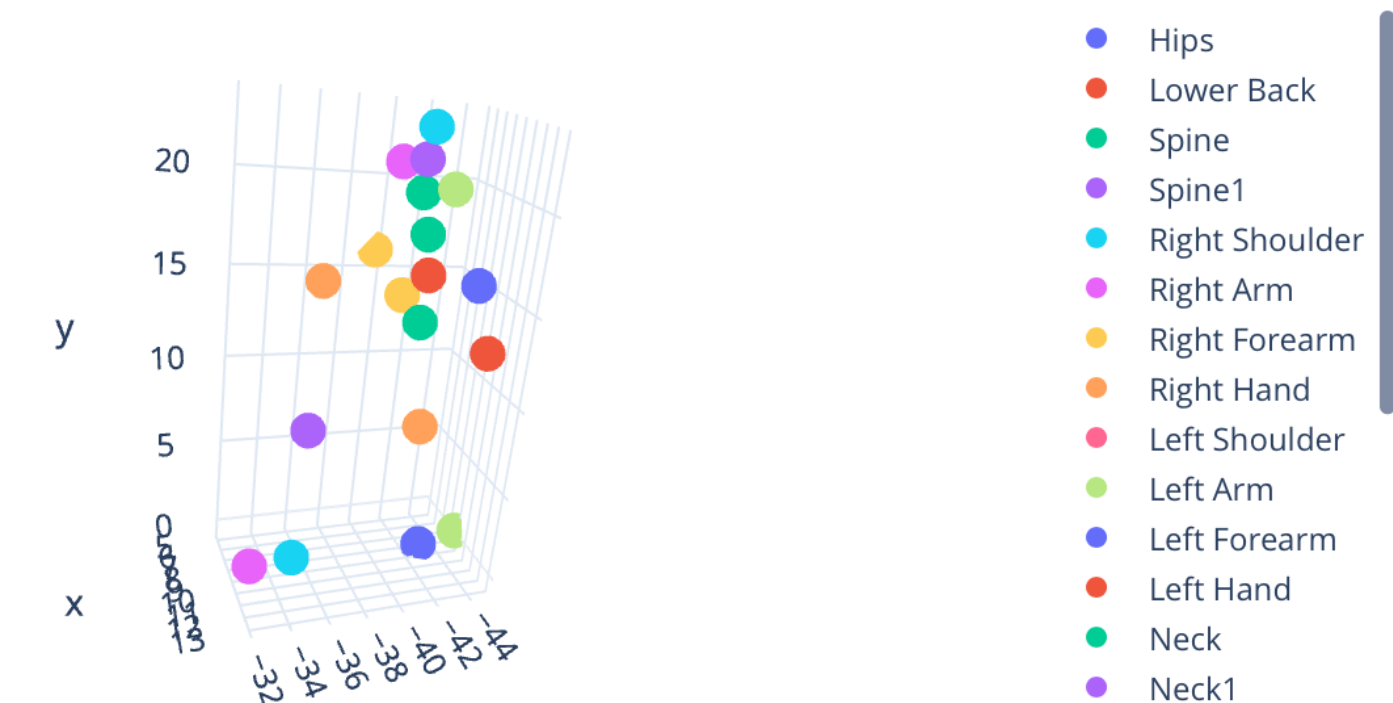
Motion capture and Eye Tracking

Motion capture data

- Recorded via
 - MoCap cameras - excellent, multiple types
 - Video - ok, issues
 - IMUs - ok, some disadvantages

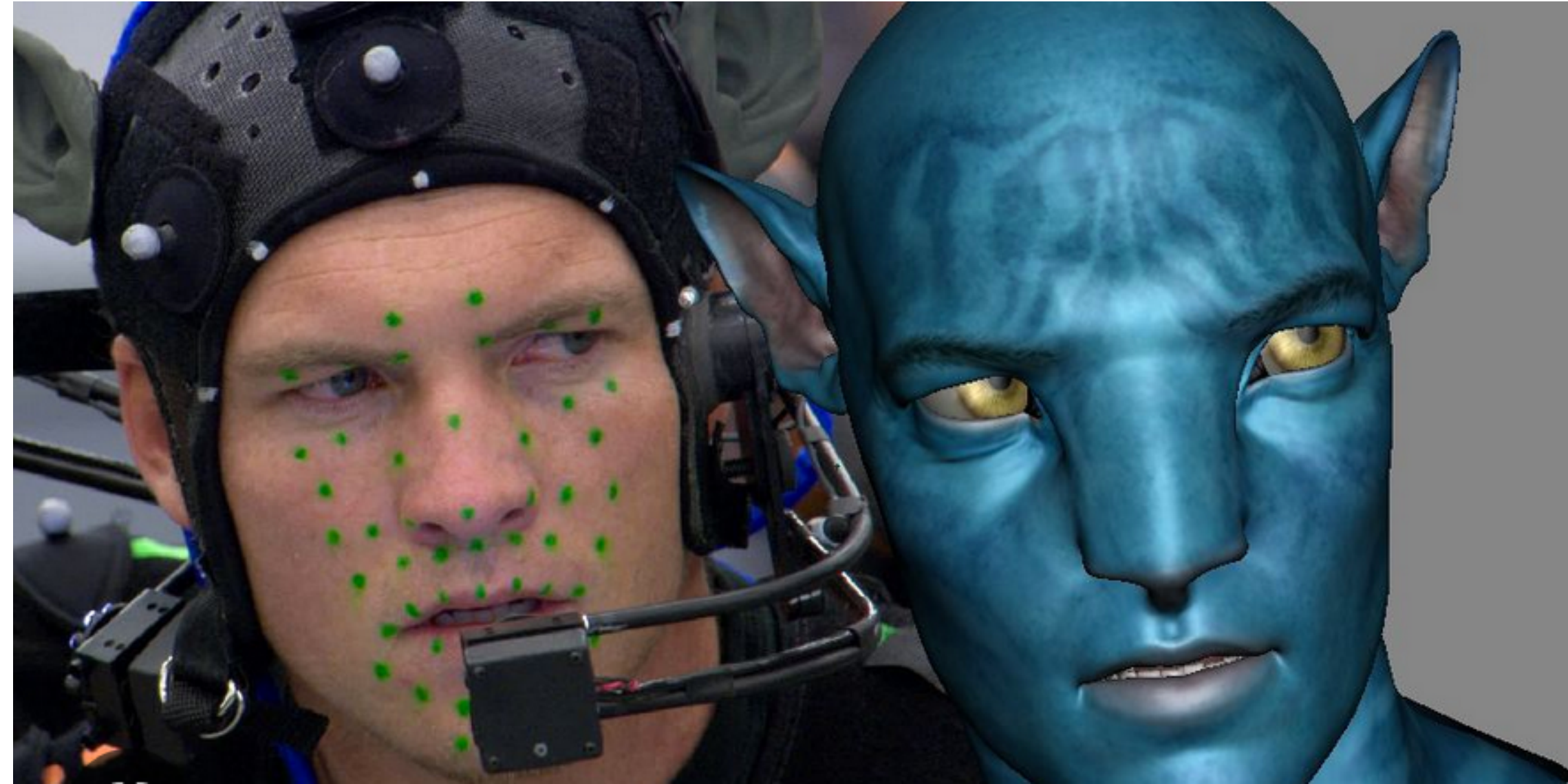


(Source: <https://medium.com/swlh/movement-classification-b98614084ec6>)



Facial motion capture

- Facial expression capture
- Using markers and a fixed perspective camera tracking with the subject
- Combined with positional markers for mapping



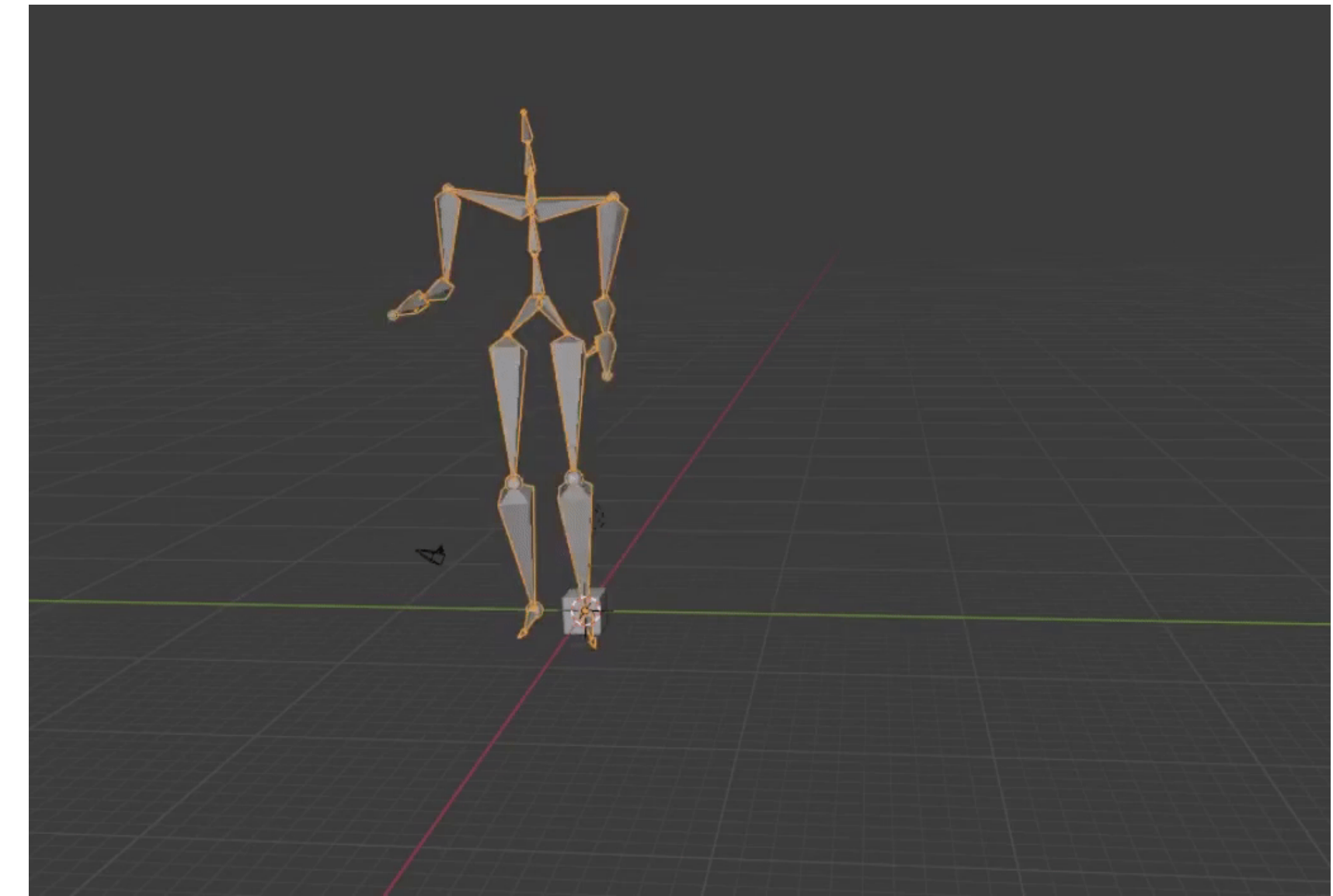
Motion capture data

- PyMO - <https://omid.al/projects/pymo/>

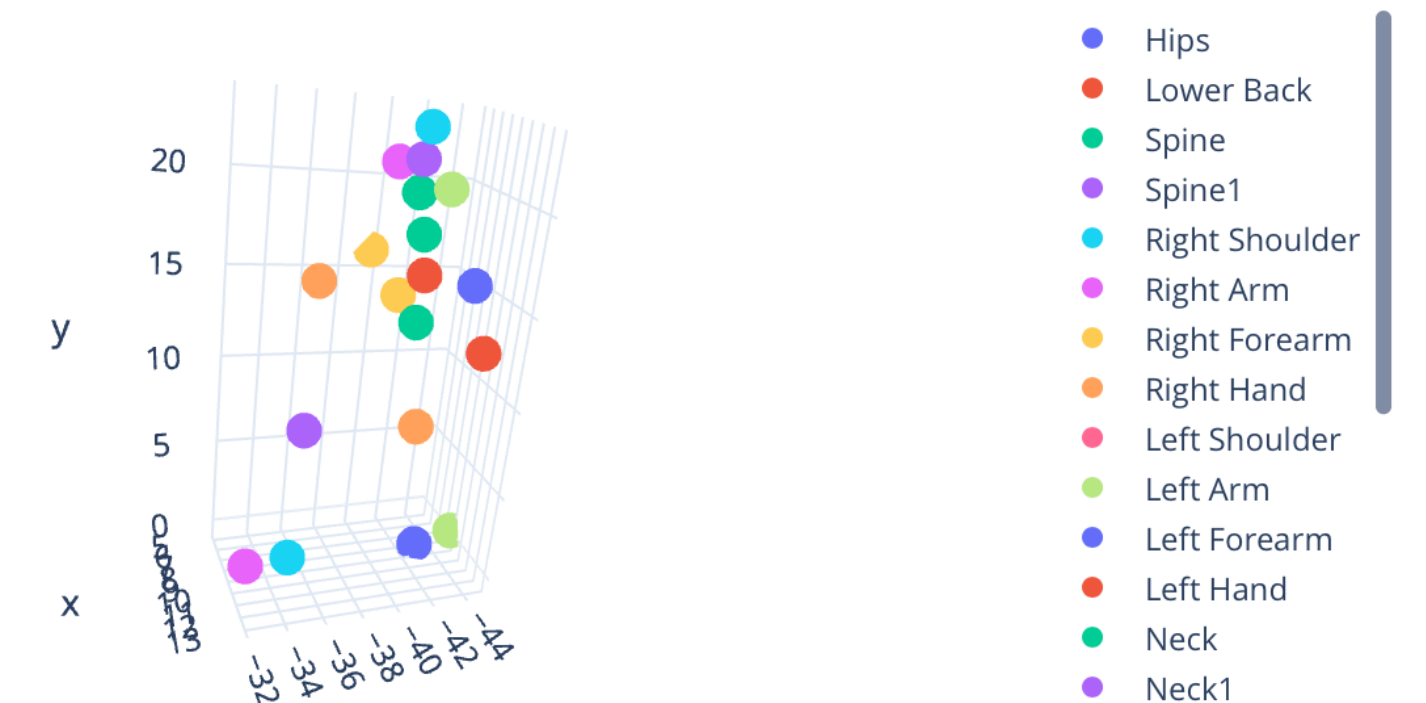
- pypi - <https://pypi.org/project/mocaplib/>

- Others

- Not well standardized yet

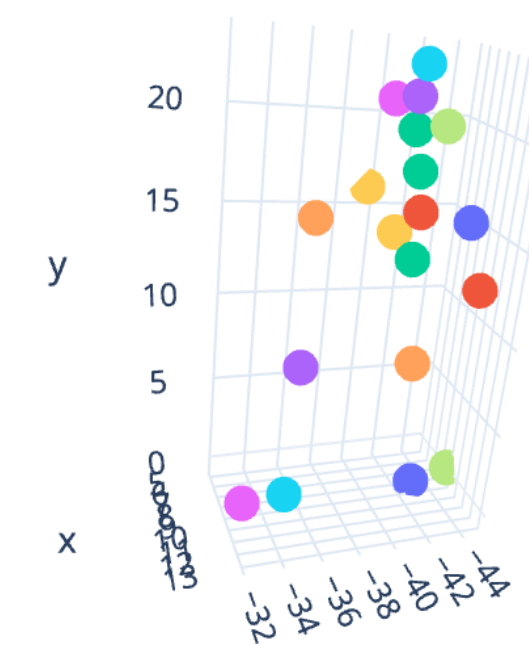


(Source: <https://medium.com/swlh/movement-classification-b98614084ec6>)



Motion capture data - challenges

- Hand manipulation involves many occlusions
 - Estimation
 - High camera density
 - Active markers
- Predictive estimation
- Marker occlusions generally, jumps and discontinuities, open/closed chain complexity
- Active systems require power, wires, may be delicate
- <https://www.engadget.com/2018-05-25-motion-capture-history-video-vicon-siren.html>



- Hips
- Lower Back
- Spine
- Spine1
- Right Shoulder
- Right Arm
- Right Forearm
- Right Hand
- Left Shoulder
- Left Arm
- Left Forearm
- Left Hand
- Neck
- Neck1



Motion capture systems

- Two main branches of tech:

- **Inertial** - IMUs track p/v/a (estimating p typically but can measure angle via gravity)

- Lower cost

- **Optical** - typically track markers, active or passive in IR to highlight marker positions relative to other data

- Higher cost

- Two main optical approaches

- **Active** systems

- **Passive** systems

- Combinations are possible



Motion capture systems

- **VICON:** <https://www.youtube.com/watch?v=HBD6vA0Xi6Y>



- **PhaseSpace:**

- <https://www.youtube.com/watch?v=A1BrYmC1Vpo>



- <https://www.youtube.com/watch?v=iklXUxpq-T4>



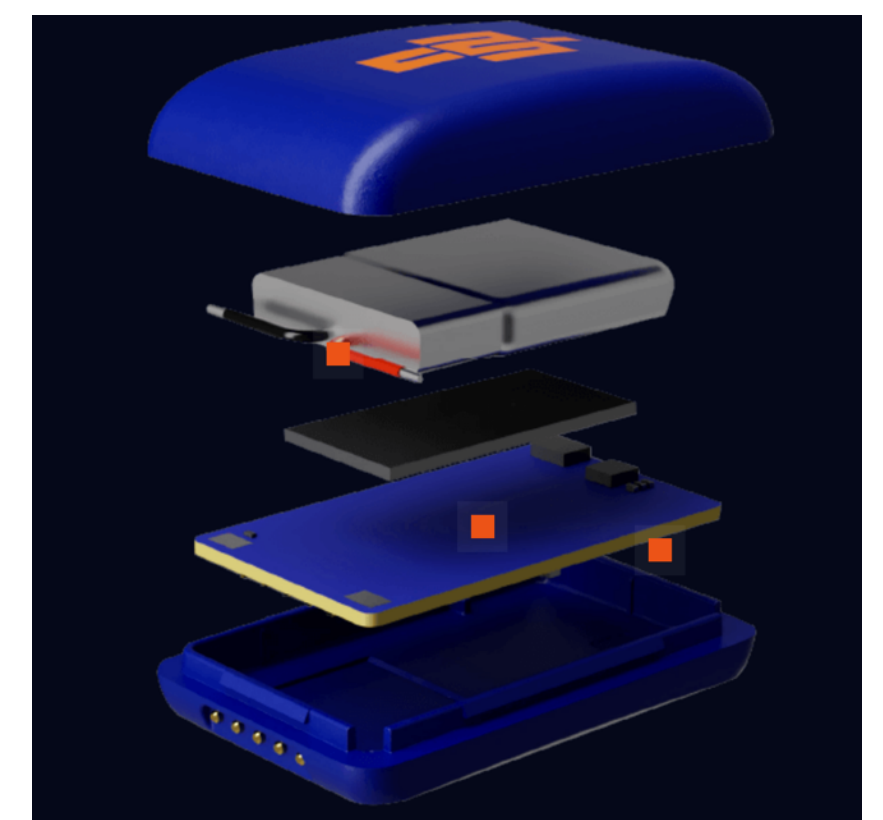
VICON

- Passive, IR camera type
- Slower refresh, non-unique markers, dependent on larger numbers of cameras for occlusion detection and reliability for complicated kinematic systems
- Works fairly well for less complex kinematics and no occlusions
- Fairly accurate at 0.017mm max
- Provides an actual video image (low res grayscale) as well potentially
- Integrated software and calibration 'wand'
- Integrated with other hardware and items like IMU
- Various software options from Vicon
- <https://www.vicon.com/applications/engineering/>
- <https://docs.vicon.com/display/Shogun18/Getting+started+with+Vicon+Shogun>
- <https://docs.vicon.com/display/Shogun18/PDF+downloads+for+Vicon+Shogun?preview=/174784515/174785494/Python%20scripting%20with%20Vicon%20Shogun.pdf>

Cameras:



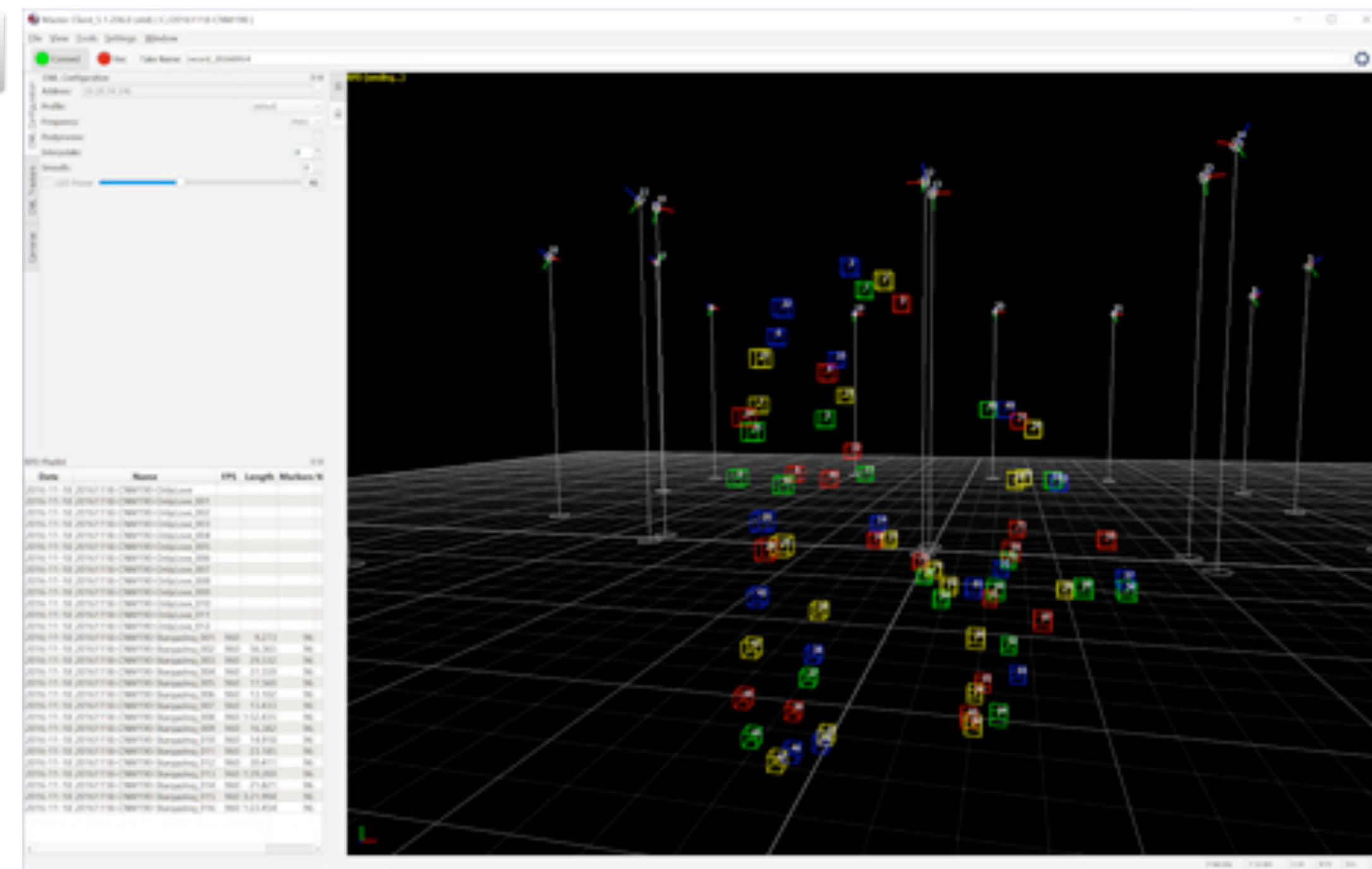
IMU:



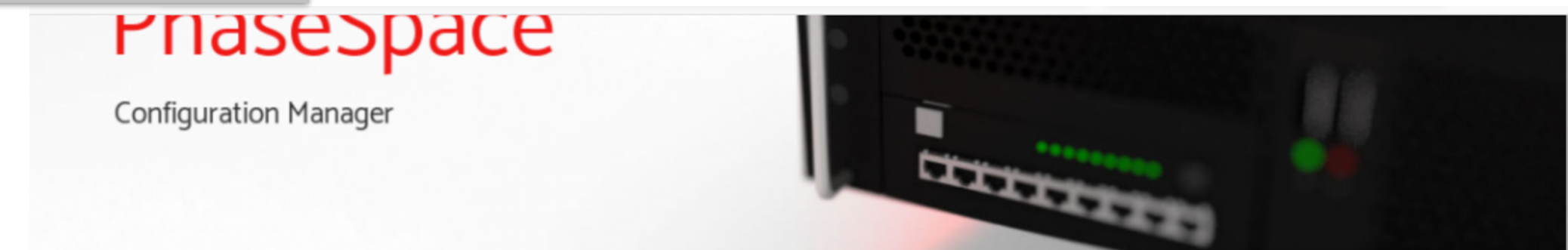
PhaseSpace

- Much faster (960Hz)
- High precision sub-millimeter precision ($\sim 20\mu$) at full sub-pixel resolution of 36000 x 36000
- No confusion between markers as each is unique, active pattern
- No video image, less cameras for reliability required
- Integrated software and calibration 'wand'
- <http://www.phasespace.com/software.html>
- <https://www.phasespace.com/applications/robotics/>
- <https://www.phasespace.com/applications/sports-medical/>

Hub:



Config. Manager:



Let's Get Set Up!

New users should visit each page below in-order.

Help is available [here](#) and by clicking the [i](#) buttons throughout the site.

Hub & Cameras
View the list of Cameras and reset the Hub.

Cameras: 6

LED Devices
Discover, monitor, encode and manage LED Driving devices.

Drivers: 1
Microdrivers: 6

Session Profiles
View and configure how LED Devices will be programmed during capture sessions.

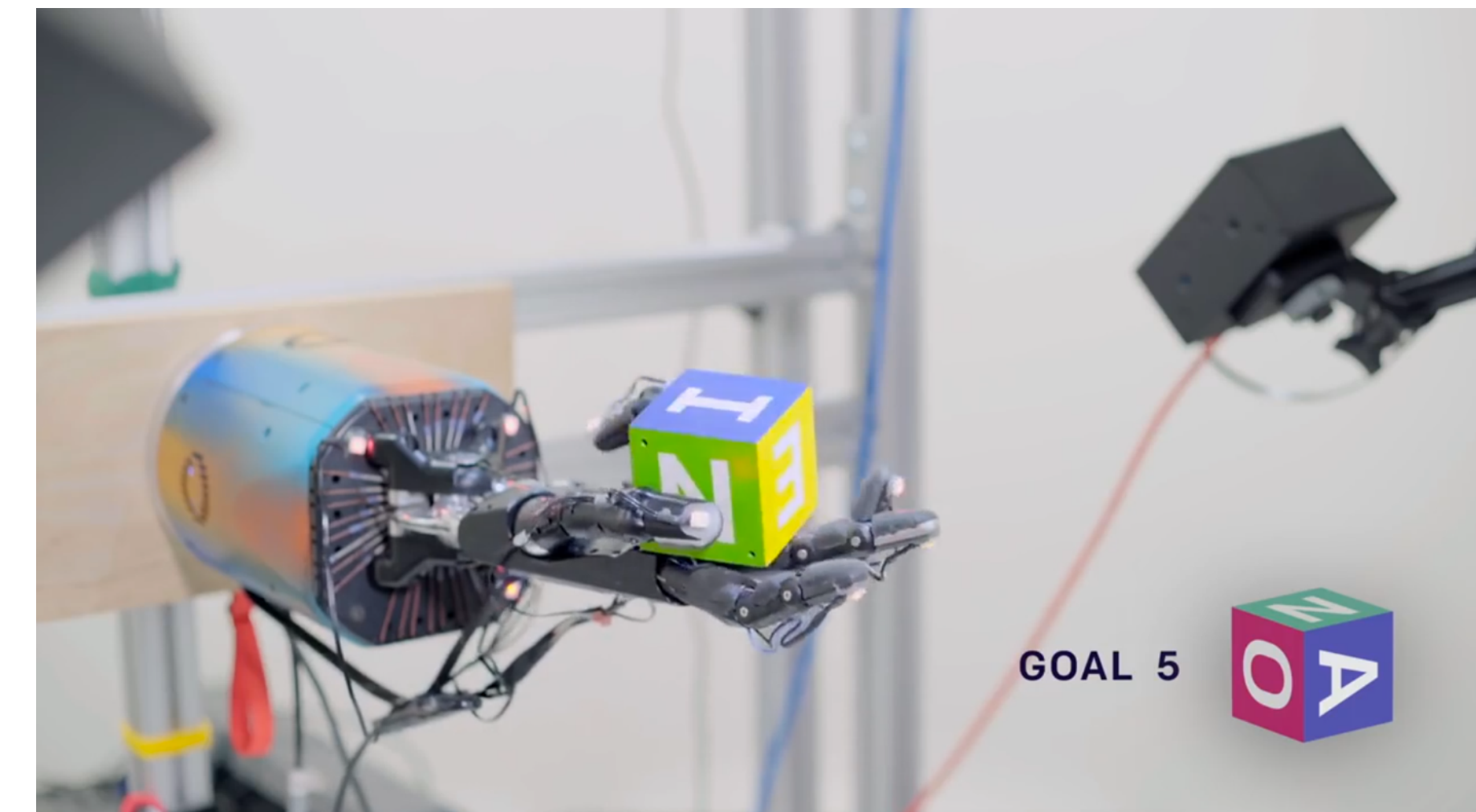
Profiles: 4

Web Clients
Calibrate cameras, Align the space and View 3D markers from your web browser.

Last Calibrated: Jun 27 2019 2:35 PM

Motion capture systems

- Dextrous manipulation
 - Neither system is perfect
 - Better to have some glove and instrumented objects
 - Many of these are not perfect
 - Relative joint angles, glove covering or interfering with movement and interaction
- Not typically suitable for MRI type studies but EEG yes
- <https://hub.packtpub.com/openai-reinforcement-learning-giving-robots-human-like-dexterity/>



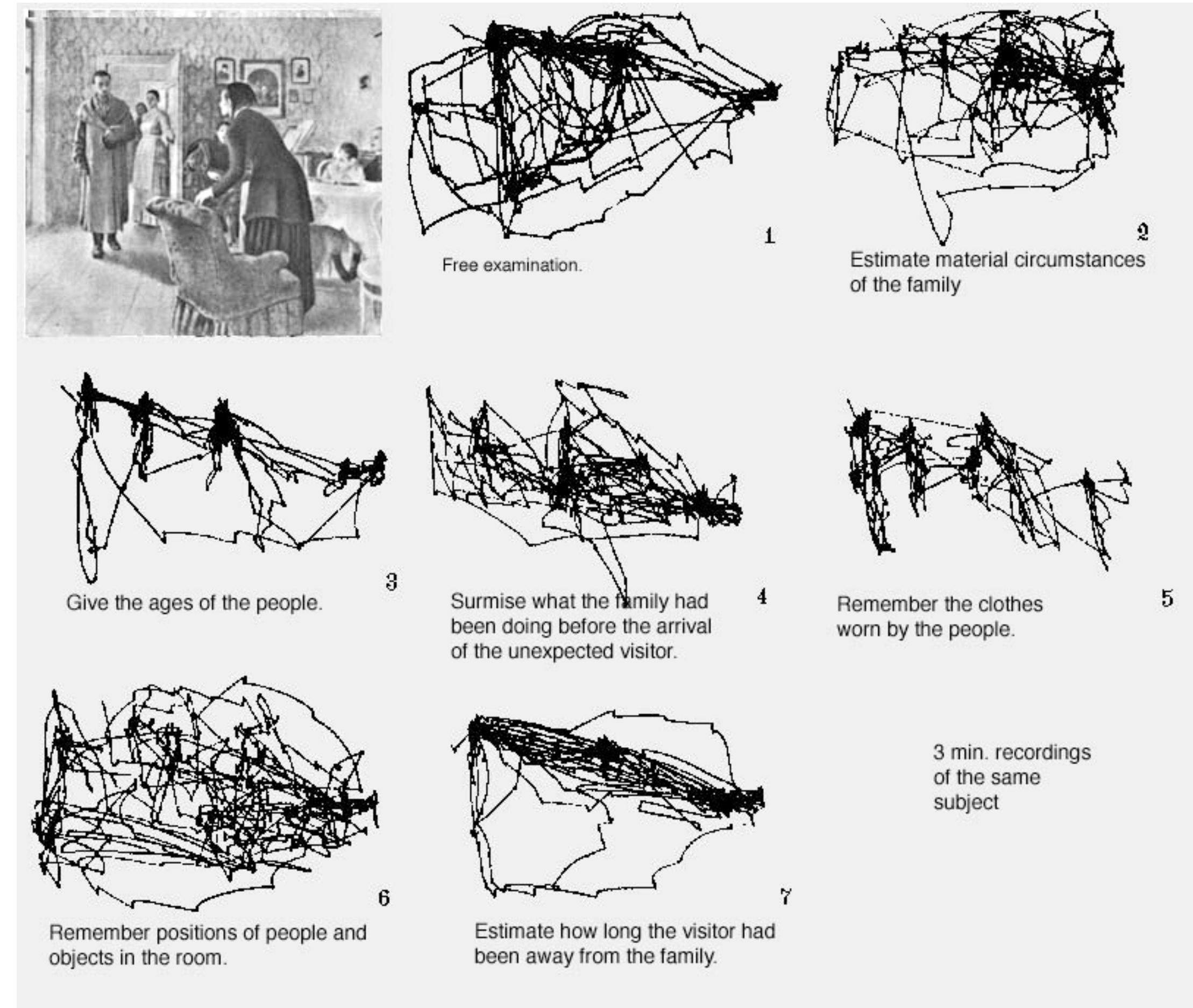
Motion capture data

()

- File type examples
 - https://en.wikipedia.org/wiki/List_of_motion_and_gesture_file_formats
 - VICON: <https://docs.vicon.com/display/Shogun17/Post+-+File+types>
 - PhaseSpace: export .C3D or .BVH files

Eye tracking

- Human eye movements are complex and indicate many things about cognitive and neurological states as well as dynamics
- Yarbis (1967) - task given to a person affects eye movement
- “Unknown water balloon release time”
- Pencil stuck in the ceiling tile going to fall but when?
- Eye position, pupil dilation indications, focal point



Eye tracking - applications

- Cognitive loading
- Neurological diagnosis
- HCI
- Language reading
- Human factors/ergonomics
- Marketing research
- Operating interfaces without other means
- Safety, game theory, aviation, other assistive applications, augmented systems, engineering, automotive, etc



Eye tracking technology

- Eye trackers use one of the following to track retinal position and other bio-optic parameters
 - Cameras
 - Electrodes
 - Eye-attached technology (special contacts etc)
- Low speed vs. high speed
- Historically way back to 1800s by observation
 - Saccades

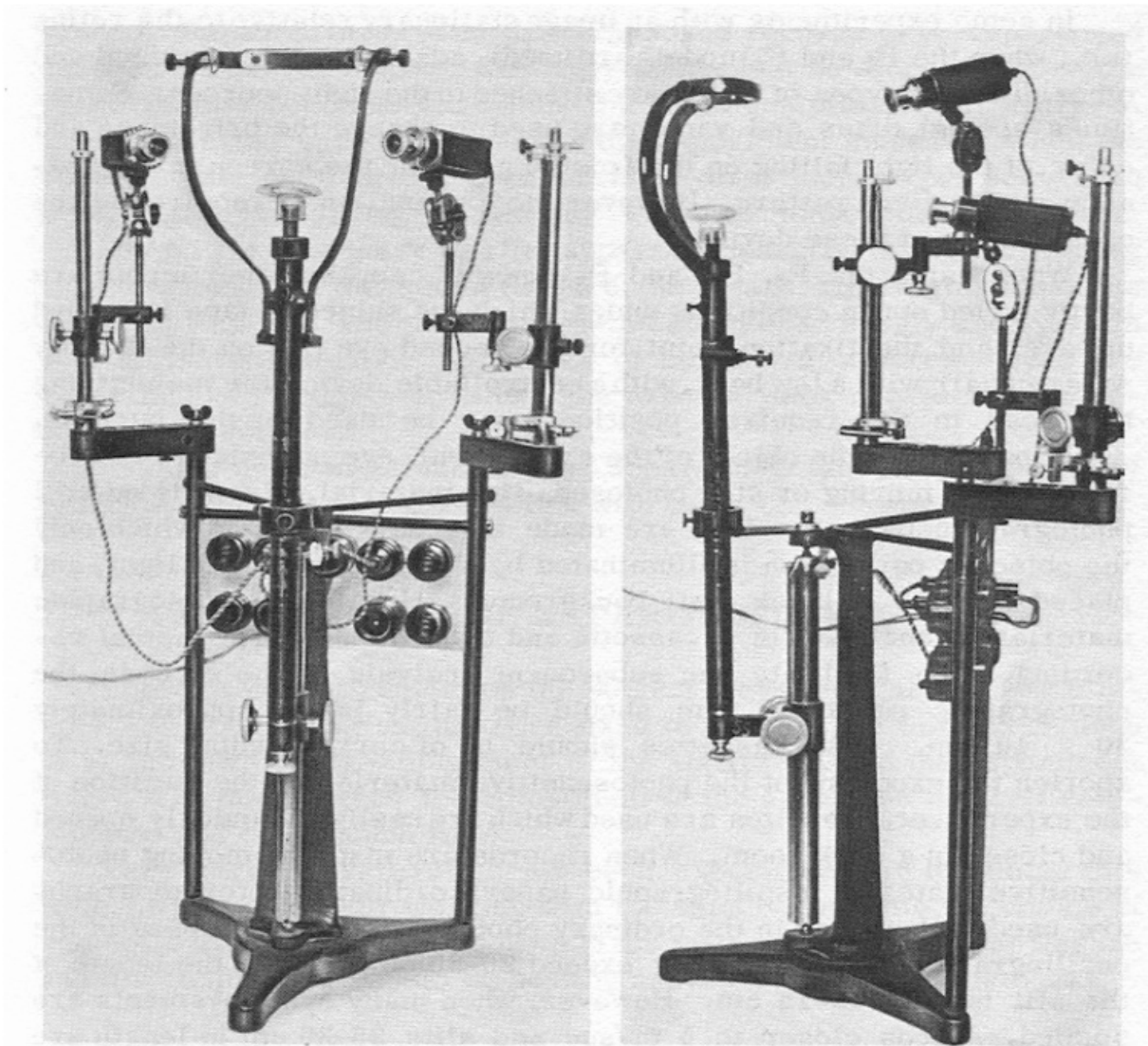
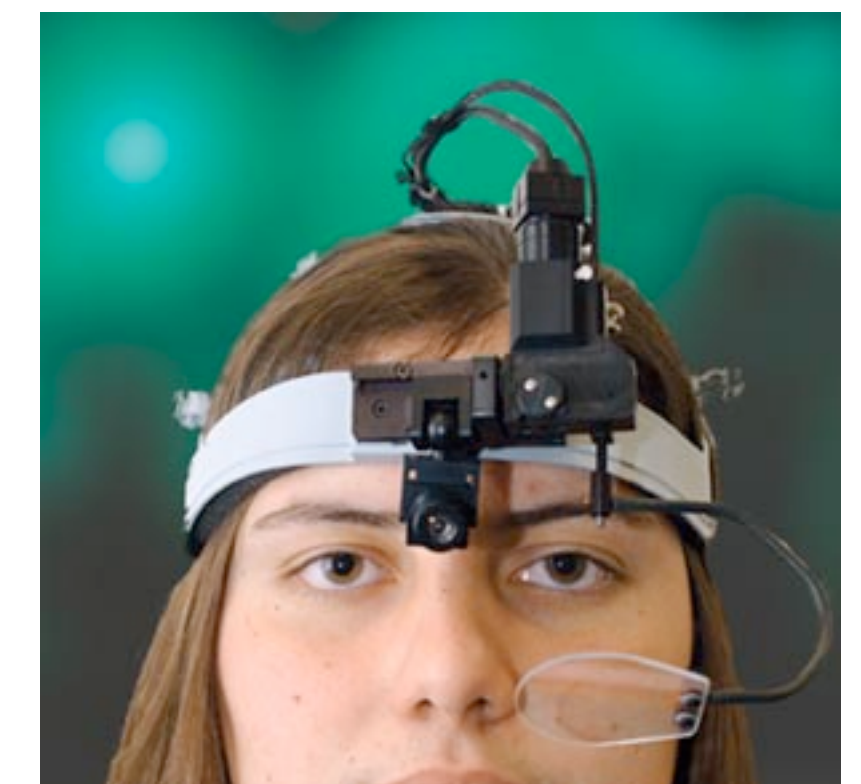
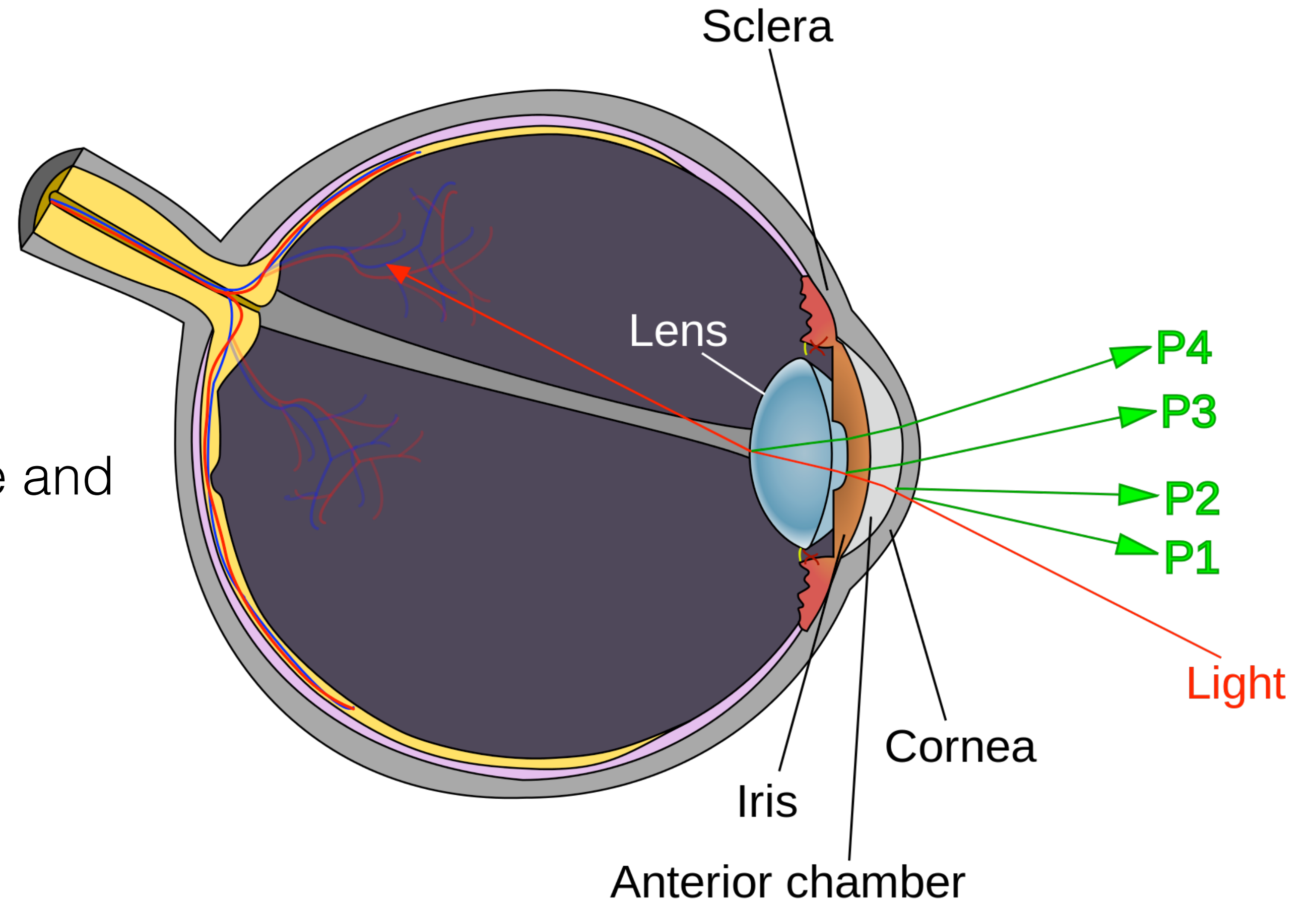


Fig. 21. The apparatus used in recording eye movements.



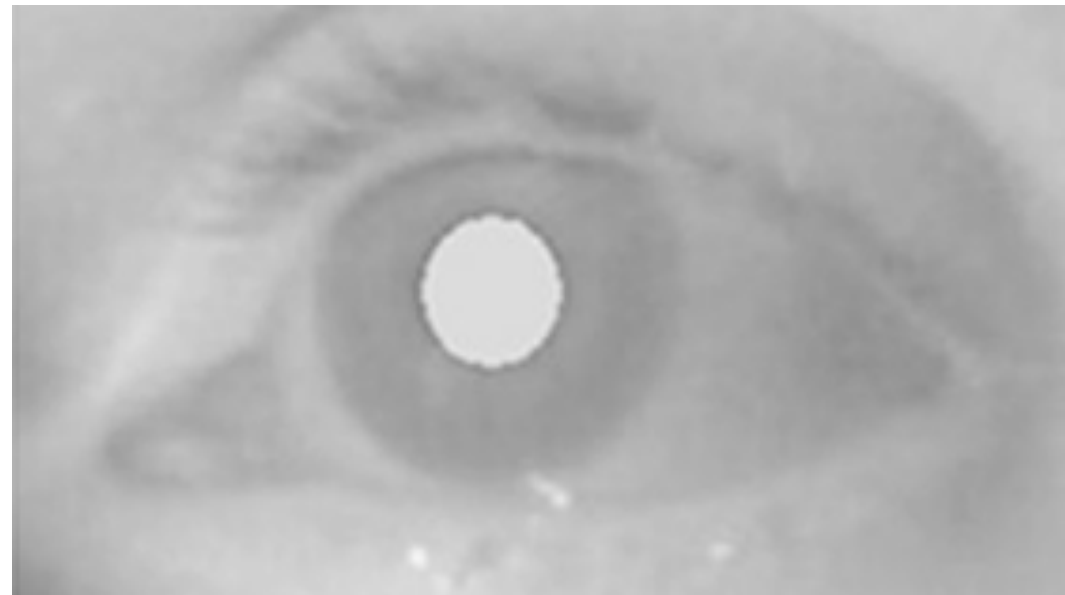
Eye tracking technology

- Most often video-based
 - Simpler, quicker to connect patient, less complications, direct measures
- Measures often infrared light reflected from eye and detected by a special camera
 - Data inferred by changes in reflections
 - i.e. Purkinje image (P1 and P4 typical)
 - or optic features like retinal blood vessel patterns

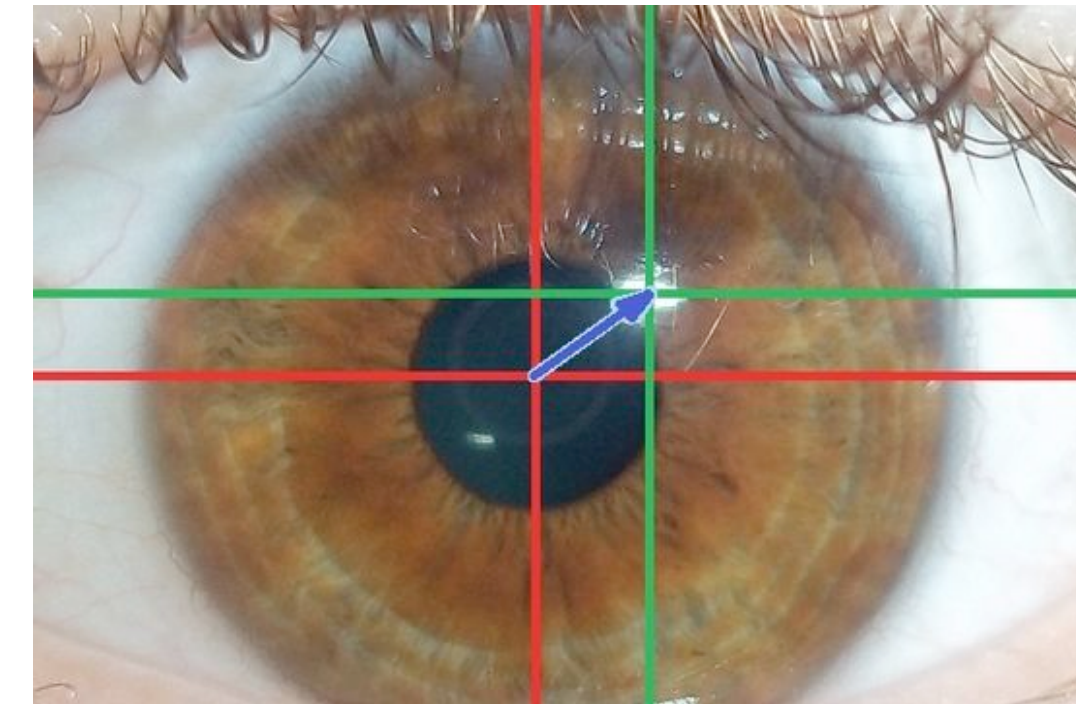


Eye tracking technology

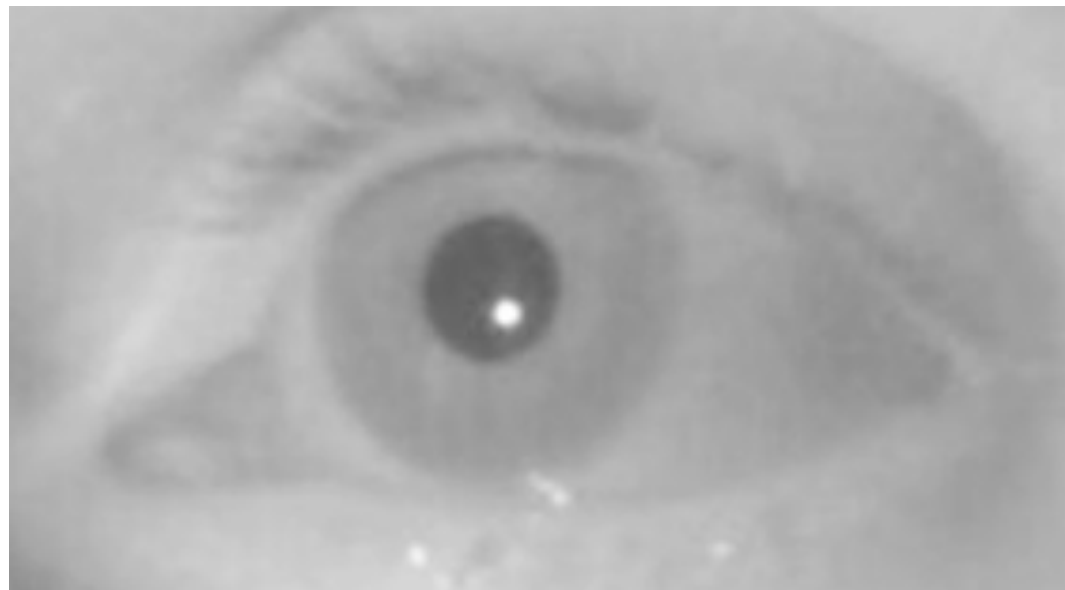
- IR/near-IR: Bright pupil,



- Visible light: center of iris (red), corneal reflection (green), and output vector (blue)



- IR/near-IR: Dark pupil & corneal reflection



Cheaper - 30Hz
But commonly > 1.3kHz

Eye tracking data representation

- Animated representations of a point on the interface
- Static representations of the saccade path
- Heat maps
- Blind zones maps, or focus maps
- Saliency maps

Eye tracking data sets and one example of raw data

- <https://www.eyetracking-eeg.org/testdata.html>
- <https://github.com/dvlastos/eye-tracking-data>
- <https://englelab.gatech.edu/dataprep/eye-tracking-data.html>

Genes and text, LISC

- Leveraging NLTK for research like gene expression studies
- Creating gene dictionaries
- Looking through literature to collect information about topics of interest, data and results using python (LISC)

Gene expression studies

- Gene expression definition
- Why animal models?
 - We use animal models for gene expression because, unless a human is undergoing brain surgery where tissue can be sampled, we cannot measure gene expression in the brain otherwise
 - Animals are found that have certain genomic similarities and assumptions are made about mapping behaviors and gene patterns into insights about humans
 - Often an animal is bred for the study with specific genes or “knockouts” are created with certain genes removed in order to understand effects
- Next time: How to use this data for neural data science

Formulating Data Science Questions

When you and your group sit down to figure out what you're going to do for your final project in this class, you'll have to formulate a strong question - one that is specific, can be answered with data, and makes clear what exactly is being measured.

The Data Science Process

Ask an interesting question.

What is the scientific goal?
What would you do if you had all the data?
What do you want to predict or estimate?

Get the data.

How were the data sampled?
Which data are relevant?
Are there privacy issues?

Explore the data.

Plot the data.
Are there anomalies?
Are there patterns?

Model the data.

Build a model.
Fit the model.
Validate the model.

Communicate and visualize the results.

What did we learn?
Do the results make sense?
Can we tell a story?

Joe Blitzstein and Hanspeter Pfister, created for the Harvard data science course <http://www.cs109.org/>.

Hypothesis testing

-Cannot prove hypothesis

-Can only reject or fail to reject null hypothesis

-Why?

Data Science questions should...

- Be specific
- Be answerable with data
- Specify what's being measured



What makes a
question a good
question?

Specifying what you're going to measure is important

Examples of poor questions that leave wiggle room for useless answers:

- What can my data tell me about my business?
- What should I do?
- How can I increase my profits?

Examples of good questions where the answer is impossible to avoid:

- How many Model 3s will Tesla sell in San Diego during the third quarter?
- How many students will apply for admission to UCSD in 2030?
- How many students should UCSD admit in 2030 for a target class size of 50,000?

Working toward a strong data science
question

Working toward a strong data science question

Vague: How does the brain change when you have a brain injury?

Better: What neurological changes are there after a stroke?

Even better: What neurological and behavioral changes can be measured with EEG and motion capture between an average normal subject and a stroke patient who had a recent stroke that impaired motor function?

Best?