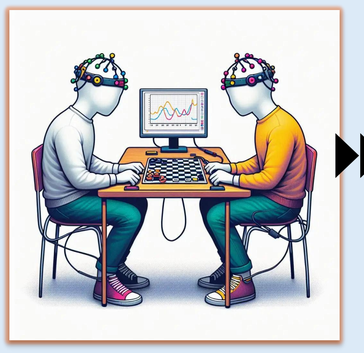
A Tool for Analyzing fNIRS Hyperscanning Using Graph Measures

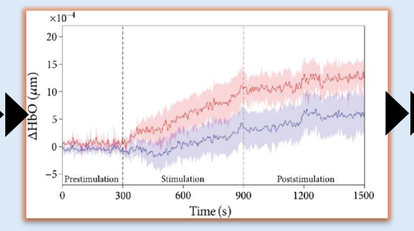
A Comprehensive Platform for Social Neuroscience Research

Software Engineering Department • Braude College of Engineering

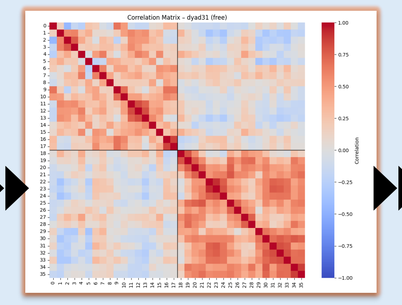
Yonatan Sherer • July 2025

| From Data to Insights

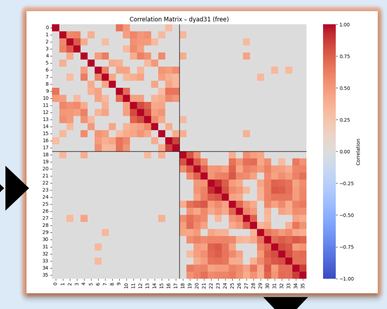




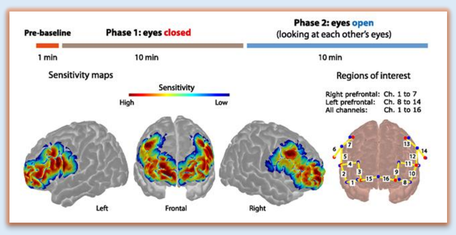
Raw Time-Series Data



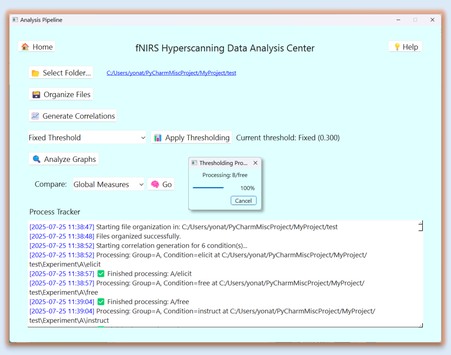
Adjacency Matrix

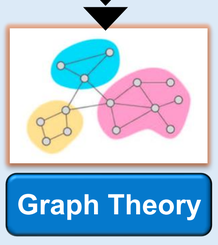


Thresholding



Experiment





Graph Theory

| Research Applications

Social Neuroscience
Studies of cooperation, empathy, and social cognition

Communication Research
Analysis of naturalistic dialogue and interaction

Clinical Applications
Autism spectrum disorder and social anxiety research

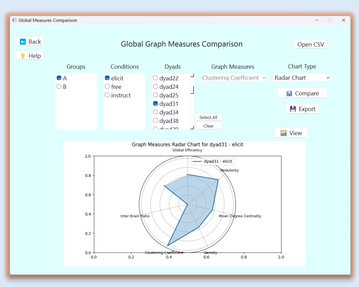
Educational Neuroscience
Teacher-student interaction studies

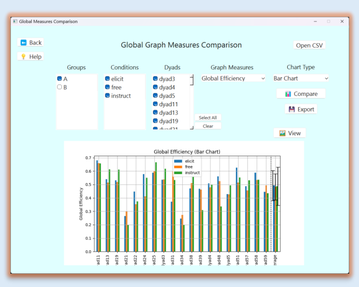
Interpersonal Dynamics
Parent-child interactions

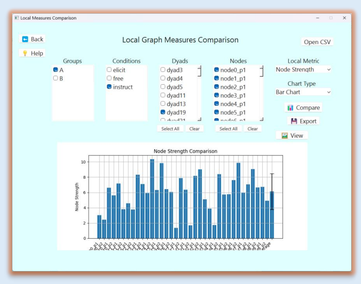
| Project Objectives

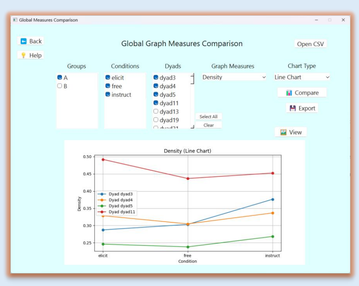
- ✓ Develop full analysis pipeline for fNIRS hyperscanning
- ✓ Apply graph theory for network analysis
- ✓ Visualize complex connectivity intuitively
- ✓ Use statistical methods for comparisons
- ✓ Design interface for diverse research communities
- ✓ Validate tool performance using real-world datasets

| Highlights









| System Architecture

Data Layer
Import and store raw data

Processing Layer
Structure input folder

Analysis Layer
Compute graphs and network metrics

Visualization Layer
Display results as plots & reports

User Interface
Interactive GUI for analysis control

| Technical Implementation

Programming

python™

Core Libraries

NumPy · SciPy · pandas · matplotlib · NetworkX Network Analysis in Python

Statistics

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

Graph Analysis

NetworkX Network Analysis in Python

Visualization

seaborn · Qt

| Key Features

- **Graph Theory Metrics** — Global & local measures
- **Automated Input Folder Processing**
- **Graph Construction with Thresholding**
- **Correlation Matrix Generation**
- **Heatmap and Chart Visualization**
- **User-Friendly Interface** — Guided workflow

| Conclusions & Future Work

- Real-time analysis capabilities
- Machine learning integration
- Multi-modal support: fNIRS + EEG
- Cloud-based processing
- Mobile application development

Contact: Yonatan.Sherer@e.braude.ac.il • **Project Repository:** github.com/YonatanSherer/fnirs-hyperscanning-tool

Acknowledgments: Special thanks to Anat Dahan and research collaborators