# Report for NeuroScope, ALMA Study 358232

Project Title:
NeuroScope: Neural Machine Intelligence Tools for Discovery and Interpretation in Complex ALMA Data

Project personnel:
Erzsébet Merényi, PI, Department of Statistics, Rice University
Andrea Isella, Co-I, Department of Physics and Astronomy, Rice University
Joshua Taylor, PhD student, Department of Statistics, Rice University
Adolfo Carvalho, MS student, Department of Physics and Astronomy, Rice University

Collaborator (not payed from this grant):
Maxwell Hummel, PhD student, Department of Physics and Astronomy, Rice University

Short-term contributor:
Patrick O'Driscoll, PhD student, Applied Physics Program, Rice University

Project period: 10/2/2017 – 9/30/2018 plus no-cost extension 10/1/2018 – 6/30/2019
Project URL: https://neuroscope.rice.edu

## Contents

# 1 Project Summary (from original proposal)

## 1.1 Abstract

With the spectacular advances of observation capabilities, the current widely used data interpretation tools often fall short of fully exploiting the rich details in data such as produced by ALMA. We propose to advance analytics and visualization capabilities for the usability of ALMA data and, in general, for complex radio astronomy data cubes.

We aim to assess, and demonstrate the effectiveness of a collection of our existing analysis tools NeuroScope, for increasing science return from complex data cubes generated by ALMA, and to further develop these tools to meet specific challenges posed by the new types and levels of complexity in ALMA data (for example, in cubes assembled from multiple spectral lines). Our existing tools were developed in earlier NASA projects of the PI, to overcome limitations of traditional methods for information extraction from terrestrial and planetary hyperspectral data cubes. These tools produced new knowledge from various data sets in other domains such as Earth and planetary science, brain mapping from functional Magnetic Resonance Images (fMRI); and they show promise in preliminary analysis of ALMA data (Merényi et al. 2016, 2017). However, ALMA data is even more complex than traditional hyperspectral or fMRI data, and the nature of its complexity is different, which motivates additional development for effective ALMA-specific tools.

Our NeuroScope tools are based on neural machine learning, and in particular, on (unsupervised) neural manifold learning, which enables sensitive identification of complex cluster structures. It also provides strong support for precise (supervised) many-class classification, as well as for inference of latent variables from spectra. We propose to assess all mathematical, methodology, visualization aspects of these tools for enhancement of ALMA usability through scientific analyses of data cubes that represent different resolutions, molecular compositions, and kinematic structures, and we will advance the analytical acuity of our algorithms by tailoring them to the specific characteristics of ALMA data. This will be guided both by insight we gain from ALMA data analyses and by studying the theoretical properties of the data. As an important outlook beyond this Study, our tools will also be applicable to data cubes assembled from disparate data such as those from different telescope observations.

We will produce ALMA-specific prototype tools (algorithms and software) for data exploitation, visualization, interpretation, and use-cases with science results to demonstrate applicability and effectiveness. Upon success in advancing science return from ALMA data with the prototypes we produce here (in the PIs research environment), we will seek technology transfer of our tools (now assessed between TRL3 and TRL4) to the ALMA environment, in a follow-up project with a level of effort appropriate for that task.

## 1.2 Study objectives recapitulated in a nutshell

We list here, in key phrases, objectives we stated in our proposal.

- Use the full spectra (without prior reduction of the number of channels) as input to neural manifold learning with Self-Organizing Maps (SOMs) to retain inherent spectral cluster distinctions and discovery potential.
- Use combined spectral lines to extract knowledge simultaneously from multiple wavelength windows, assess the value of this versus finding structure from single lines.

- Automate cluster extraction from learned SOMs, with clustering quality closely approximating the quality of interactive cluster extraction. This was the main challenge, which needed heavy algorithm development and innovation. Interactive cluster extraction has shown excellent results of capturing sophisticated structure and finding interesting anomalies in ALMA cubes prior to this project. However, interactive SOM segmentation does not scale to pipeline processing. Automation is needed, without loss of the clustering quality.
- Make automated cluster extraction fast, for future pipeline use. Innovation for automation was, in part, also aiming to address algorithmic efficiency. In addition, consider parallel hardware implementation for acceleration of SOM learning.
- Make all processing repeatable, and as objective as possible.
- Develop new visualizations for viewing clusters derived from high-dimensional spectral cubes.
- Develop prototype software, provide description.
- Demonstrate the capabilities on
  - a real protoplanetary disk where planet formation is suspected, therefore departures from the Keplerian kinematics would manifest
  - a molecular cloud, the Carina Nebula
  - synthetic protoplanetary disks, for verification against known template.
- Assess and elaborate the challenges for deployment to ALMA (given the opportunity after this Study).

# 2 Accomplishments

## 2.1 Relevance to ALMA, and to Machine Intelligence

**Relevance to radio astronomy:** The exquisite sensitivity of ALMA, as well as it unprecedented capabilities of mapping astronomical sources at both high angular and spectral resolution, challenge traditional methods of data analysis which mostly rely on the expertise of the investigator and/or on casting multidimensional data onto two dimensions (e.g. using intensity moments). Although this provides an easy way to inspect complex observation products, reducing the number of dimensions of data prior to the analysis may lead to losing information. Instead, machine learning algorithms such as NeuroScope allows for exploiting multidimensional data without loss of information. However, the increased amount of detail that NeuroScope returns compared to moment maps; or the different details compared to clustering tool such as ClumpFind ([BRJE07]), pose challenges to the astrophysical interpretation of the outcomes of the machine learning analysis which require deep understanding by experts. For these reasons, in this project we both developed the software tools necessary to process ALMA data with NeuroScope, as well as use our expertise to investigate the astrophysical relevance of the results.

**Relevance to Machine Intelligence:** NeuroScope has three major elements that aim to enable smart, automated and fast processing for discovery of complex structure in large spectral data cubes. "Precision" cluster finding from learned Self-Organizing Maps (SOMs) with an interactive tool (using the highly compressed representation of the data by the sparse connectivity (CONN) matrix derived

from SOM knowledge [TM09]) has been demonstrated by us before this Study. Acceleration of SOM learning also existed. Automation of cluster extraction with comparable quality and clustering detail to that produced interactively by human experts did not exist for large, complex data; only existed as our early experiments. In this Study we developed two major procedures that eliminate this bottleneck and open the way to repeatable mass processing. These are described in 2.4 and illustrated in 2.5, and are in published papers [MTI17, MT19, TM19]. Both use modern graph-segmentation algorithms to segment the CONN matrix after performing further sparsifications on it. The results are comparable to interactive segmentations and are very fast (typically much less than one second for one clustering). In contrast, the same graph-segmentation algorithms take prohibitively long time, and produce extremely poor results when applied directly to the data (not to the SOM / CONN learned from the data). We have not seen other work that accomplishes automated segmentation of complicated high-dimensional data — such as ALMA cubes — from learned SOMs with the level of detail and quality, as well as high speed, as our automated procedures. Once mature, they will enable automatic generation of clusterings in a pipeline aimed at producing science ready data products.

## 2.2 Accomplishments with respect to project goals (algorithm and software development, application to data analysis)

The following short paragraphs summarize our accomplishments in response to the study goals listed in section 1.2. Software modules referenced (by names in **`bold face courier fonts`**) are reviewed in Fig. 1, as well as detailed in section 2.4, and in slides and/or documentation attached in section 4. Sample analyses are given in section 2.5, and in slides attached in section 4.2.

### 2.2.1 Use full spectra and combined wavelength regions for analyses

NeuroScope is fully capable of using all spectral channels of an ALMA cube, or all channels of combined multi-line cubes. In this study, the comparison of clusters (areas of similar kinematic behavior) based on all channels of single $C^{18}O$ and $^{13}CO$ lines with each other and with clusters from combined $C^{18}O$ and $^{13}CO$ lines show increased discovery power from the combined lines; and also other interesting results (see 2.5).
In this Study we clustered data cubes along the velocity/spectral dimension, which means that the clusters are regions of similar kinematics. However, the same technique could naturally be applied to clustering the data along one spatial dimension, which would lead to clusters with a different meaning. This could be something to explore in the future.

### 2.2.2 Automate cluster extraction from learned SOMs, with details comparable to interactive segmentation by human expert

This was the most important and most challenging goal of the Study. We achieved and surpassed it. It is a unique accomplishment in that we do not know of other automated approach to the segmentation of a Self-Organizing Map (SOM) that produces the level of precision and details from SOMs of complex-structured data as our new procedures. Details, including an overview of previous SOM-segmentation work, are given in ([MT17, MT19, TM19]). We developed two major software modules, **`gsegSOM`** and **`DM-Prune`**, both based on using the CONN (connectivity) graph derived from the learned SOM as input to leading modern graph-segmentation algorithms (we denote this by "CONN + igraph" since we use the igraph package implementation of graph-segmentation algorithms). Both utilize available parametrizations (thresholdings of CONN, and algorithm parameters). The difference between the

two modules is that **gsegSOM** takes a pre-specified grid of parameters given by the user, produces a clustering for each grid point and subsequently ranks the clusterings for recommendation; while **DM-Prune** optimizes CONN parametrization (pruning of graph edges) based on information loss, thus narrows the recommended clusterings to a small handful. Sections 2.5 and 4.4 elaborate on these, with illustrations.

### 2.2.3 Make SOM learning and automated clustering fast

As part of the automation of cluster extraction, algorithmic efficiency was sought. One clustering of a typical ALMA data cube with the CONN + igraph approach takes $\ll 1$ sec, so even producing a few dozens can be done in a very short time. The remaining bottleneck is the SOM learning itself (with pre-existing NeuroScope module **annCSOM**), which is a long process in software. However, as of May 2017, the PI obtained a custom-FPGA board (RAPTOR) that implements the same conscious SOM as used in NeuroScope (an updated version of [LMPR13]) and exercised it in the context of another project aimed at knowledge discovery from remote sensing spectral imagery. A properly learned SOM of ALMA cube stacked from $C^{18}O$ and $^{13}CO$ lines of the protoplanetary disk HD 142527 (sec 2.5) can be produced in 5 - 10 seconds with this medium capacity board. This existing capability (currently outside of this Study), or further improved higher-end version with 4 times or higher speed, is available for pipeline integration, see 2.5.

### 2.2.4 Assure repeatability of processing

All NeuroScope modules are designed to take a pre-filled work order (worksheet); and they produce a process log file, in addition to many intermediate products that serve for verification. These record input / output file names, process parameter settings, etc. so that every run can be traced and repeated. Exceptions are the interactive modules where user actions (keystrokes / clicks) are not recorded; however, this can be done if desired. The paired down MATLAB implementation of core NeuroScope functions, **NeuroGlimpse** is somewhat limited in this respect (see section 2.4).

### 2.2.5 Develop SOM-sepcific and ALMA-specific visualization and evaluation tools

Visualizations for SOM evaluation and for cluster viewing / evaluation are part of pre-existing modules **annCSOM** and **remap**, while new module **gsegSOM** offers similar and many additional details, integrated with automated SOM segmentation. These are described and illustrated in sections 2.5 and 4.4, as well as respective user documentations in the Appendix.
The module **specter** is designed for viewing the spectral signatures from a selected spatial window of an image cube, as a plot array, and perform simple manipulations. The spatial window is selected from a reference image that represents the spatial dimensions of the image cube; for example, a continuum image; a single channel map; or a color cluster map. When the reference image is a cluster map, additional, cluster-specific operations are available. Implementation of physically meaningful, intuitive coloring schemes for cluster maps was a study goal delayed by unplanned forced development items (section2.3). However, it is currently under development, and will be done in the next month or two. Sections 2.4 and 2.5 provide details.

### 2.2.6 Develop prototype software, provide description

Development of new prototype software has exceeded our goals, with the exception of intuitive color schemes for cluster visualization (as in section 2.2.5). We have not one, but two new algorithms for

automation of SOM segmentation / clustering with quality close to that obtained by human expert interactively, as in 2.2.2; novel SOM / data cube / cluster visualization; crafted product exchanges between CASA and NeuroScope, and a novel way to compare 3D ClumpFind and 2D NeuroScope clustering results. Please see details and use-case illustrations in sections 2.4, 2.5 and appendices.

#### 2.2.7 Demonstrate NeuroScope capabilities on ALMA data

Section 2.5 and appendices 4.2, 4.4 and 4.4 provide illustrations of capabilities through use-case examples for the HD 142527 protoplanetary disk, the Carina Nebula, and synthetic disks. Published papers([MTI16, MTI17, MIT18, HTM$^+$19] give more details on ALMA data analyses. Published technique papers ([MT17, MT19, TM19] show clustering studies with additional, synthetic and real terrestrial spectral images similar in complexity to ALMA cubes, but for which ground truth is available to formally assess the validity of the clusterings.

#### 2.2.8 Assess the challenges for deployment to ALMA

This Study proposed to produce and demonstrate prototype capabilities, without delivering software. (The exception is a paired down MATLAB version of core learning and visualization functionality, see **`NeuroGlimpse`**, section 2.4.) The challenges involved in deployment of full NeuroScope capabilities to ALMA vary by components. These are detailed in section 2.6.

### 2.3 Forced unplanned efforts essential to the Study

#### 2.3.1 Combination of ALMA single dish and interferometric data

A key part of this Study involved the analysis of ALMA observations of a 60"$\times$30" photo-dissociation region (PDR) in the Carina Molecular Cloud obtained as part of project 2015.1.00656 (PI: P. Hartigan). The observations were performed at Band 6 (1.3 mm) and covered the $^{13}$CO and C$^{18}$O (2-1) lines, as well as the continuum emission. To recover the large scale line emission and kinematics, interforometric observations obtained with both the 12m array and the ACA were combined with ALMA Single Dish (SD) data. The combination was first attempted using the "feather" tool in CASA 4.7 following the instructions in the CASA "User Reference & Cookbook". However this led to unsatisfactory results characterized by image artifacts. As an alternative approach, we performed the data combination using the "tp2Vis" CASA tool (https://github.com/tp2vis/distribute) which first calculates visibilities from SD images and then combines them with the interferometric data to generate a combined visibility file. In doing that we discovered that none of the SD deconvolution algorithms developed by the "tp2vis" team produced satisfactory SD models. The success, or failure, of SD deconvolution was assessed by convolving the SD model with the SD primary beam and comparing the result to the original map. We tracked down the problem to the fact that "tp2vis" performs SD deconvolution in the Fourier domain which leads to severe aliasing if the SD image is characterized, as in our case, by very diffused emission that covers most of the field of view. To overcome this problem, we developed our own deconvolution algorithm which operates in the image domain following the prescription of "hogbom" deconvolution. This produces a single image, or data cube, with point-source model components of the SD data that could be fed to "tp2vis" to calculate SD visibilities. The SD-interferometric data combination problem resulted in a delay of a few months for this Study and forced us into a no-cost extension. However, the produced SD deconvolution algorithm is fully general and it could be released to the public enhancing the scientific return of the "tp2vis" package.

### 2.3.2 Challenges in generating synthetic disk models

This Study was originally designed to take advantage of our capability of generating well-controlled models for the line emission of protoplanetary disks to train ourselves in interpreting clustering maps produced by NeuroScope. To this end, we used the hydrodynamic code LA-COMPASS to generate models of disks perturbed by planets, the radiative transfer code RADMC3D to calculate the CO line emission, and the CASA task "simobserve" to simulate ALMA observations taking to account both the discrete sampling of the uv-plane and the effect of thermal noise. The resulting data cubes were then processed using NeuroScope with the goal of assessing its capability of discovering faint signatures in the line emission of perturbations induced by the planets. In particular, we focused on the possibility of using NeuroScope for detecting vortexes generated at the outer edge of gaps opened by planets based on the models discussed in [HIL+18].
Unfortunately this test led to inconclusive results. The main reason of this was the fact that the LA-COMPASS code that does not allow to generate the intermediate-level products required for a quantitative comparison with the clustering map produced by NeuroScope (e.g. a map clearly showing the position of the vortex). Furthermore, the analysis was complicated by the fact that a vortex does not produce well defined signature in the CO line emission, and by the fact that planets induce a myriad of disorganized perturbations in the gas kinematics. Although useful for the general understanding of planet-induced perturbations, these test resulted in unplanned efforts and in the loss of time.

## 2.4 Software prototypes developed in this Study

Boxes in Fig. 1 portray our capabilities, with color codes for pre-existing NeuroScope tools (boxes in light blue background), and for tools developed during this Study (boxes in magenta background). Further distinction is made between mature software (medium blue boxes) and fully functioning new software that is in maturation phase (dark orange boxes). Light yellow boxes indicate components in testing phase. Fig. 1 also illustrates the phases of tool development through the typical flow of data analysis steps. A bank of component modules in each of the columns of steps 1 - 6 are possible choices to realize the respective action indicated above the step number.
Below we describe software developed in the course of this Study. These have magenta background in Fig. 1. In addition we mention the role of some pre-existing (blue) components for context, and provide documentation for the flagship module **annCSOM**. We list the components in the order they appear in Fig. 1. We also provide additional details (illustrations of functions or user documentation) for key components in the Appendix (section 4) as indicated in their respective paragraphs below.

**1. Functions for NeuroScope – CASA interface**, for import / export of products:

**FITS2NeuroScope** is a MATLAB routine which digests an image cube from a FITS file into one of the formats required by NeuroScope (.viff) for SOM learning. Additionally, this routine allows quick viewing of the imported image bands (for spot-checking the integrity of the import procedure), replaces any non-numeric spectral measurements (such as missing NAs), and generates auxiliary information needed by NeuroScope routines (a table of min/max values of the data used for internal scaling of each band, a wavelength file, and, if applicable, a spatial mask so that NeuroScope can avoid processing any pixels with missing data).

**ClumpFind2DProj** is an R routine which reads the output of clusterings produced by ClumpFind (which are produced in .fits format), projects them to a 2-D image footprint, and saves the resulting clustered image in .png format for easy viewing and comparison. ClumpFind detects clusters with 3-dimensional cluster boundaries (i.e., also along the spectral axis). This routine creates a 2-d pro-

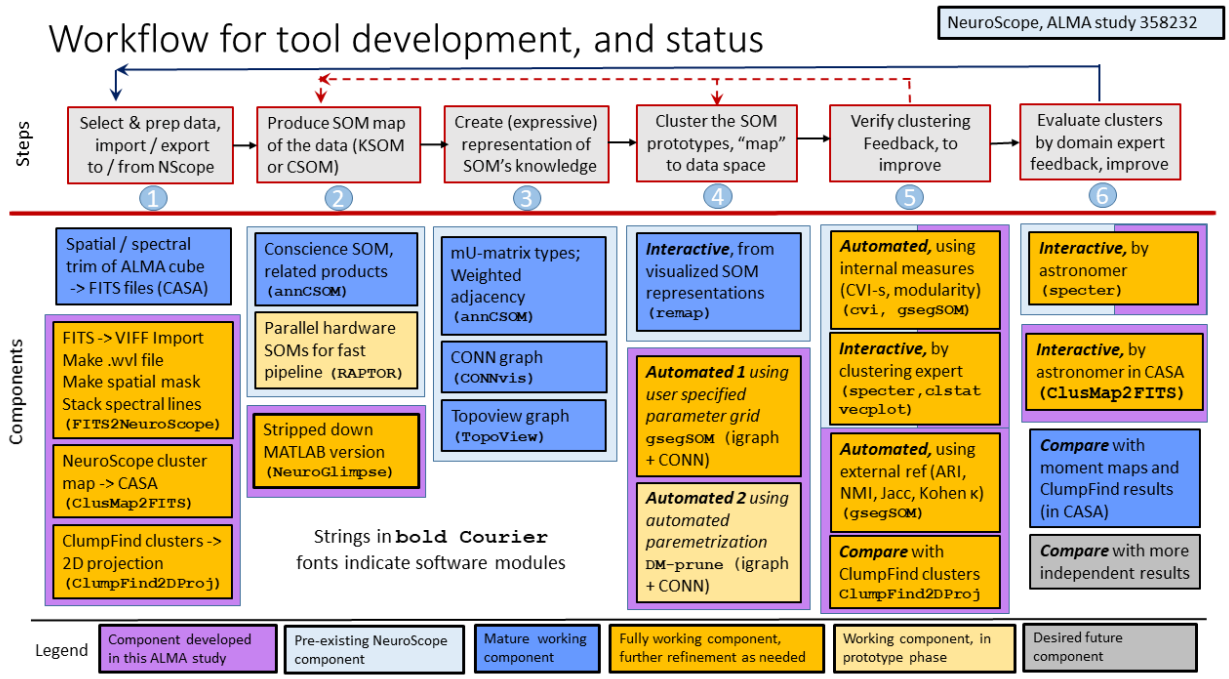



Figure 1: Top-level overview of NeuroScope capabilities, and workflow for tool development. Components developed in this Study have magenta background, pre-existing ones are shown on light blue background. Deep blue boxes indicate mature software; orange boxes signify well-exercised modules that can be further developed; and yellow boxes show components in testing phase.

jection of these cluster footprints by assigning to each image pixel the plurality cluster label which ClumpFind assigned along its bands. Colors are automatically assigned to each distinct ClumpFind label for visualization.

**ClusMap2FITS** is an R routine which reads a clustered .png image (i.e., one that is output by gsegSOM) along with the .fits header of the data file which the clustering represents. The information stored in the .fits header is copied and assigned to a new .fits file of a 1-band image, whose pixel values are integers denoting unique cluster labels in the input clustered .png image. Ensuring the .fits header information is identical in this file allows the scientist to overlay the **gsegSOM** cluster structure onto a .fits image file in CASA and use CASA's native toolkit to explore areas of quantities of interest arising from the clustering. Note that this functionality is also afforded via the use of specter.

## 2. Components to perform SOM learning

**NeuroGlimpse** is a simple MATLAB implementation of core NeuroScope functionalities (SOM-learning, visualization, generation of CONN graph, implemented in pre-existing modules **annCSOM**, **CONNvis**, **TopoView**), stripped of the algorithm development support, housekeeping, tracking, and auxiliary functions. We offered it in our original proposal to provide a taste of the NeuroScope clustering approach. NeuroGlimpse can perform CSOM learning, and generate the products needed for input to the automated SOM segmentation module **gsegSOM**. It also includes interactive clustering capability (a simplified implementation of **remap**). While we cannot deliver NeuroScope software (nor was it

planned) in the context of this Study, the automated clustering module **gsegSOM** may become available sooner than the rest of NeuroScope (see section 2.4). **NeuroGlimpse** enables an astronomer to learn an SOM and generate the products required by **gsegSOM**.

The **RAPTOR** is a custom parallel FPGA hardware that performs either Kohonen SOM or Conscience SOM learning, and produces output files that interface seamlessly with our NeuroScope software. The RAPTOR was developed by the PI's collaborators at Bielefeld University, Germany. Substantial testing of this board was facilitated by the PI. The first version is published in [LMPR13]. The PI's group has a mid-level version of the RAPTOR, which runs approximately 500 times faster than SOM learning in software on our ordinary Linux machines. As an example, this board learns an SOM of the 200-channel HD 142527 ALMA cube in Fig. 5 in about 5 seconds. Higher-capacity and faster versions (a factor of 4 in speed) of the RAPTOR are available, which also include a fast data bus, thus could serve in a pipeline. An additional attractive feature is that the RAPTOR has very low energy consumption.

**3. Components to generate SOM representations and visualizations** are partly built into **annCSOM**; or implemented in other pre-existing modules **CONNvis**, **TopoView**. All generate visualizations with default settings (such as CONN matrix thresholds for connection strength, local connection rank; (see [TM09] [MTZ09, MT19]), and can be manipulated interactively. The numerical representations can be used as input to subsequent analyses. For example, the CONN matrix is used as input to automatic SOM segmentation.

**4. Components to perform SOM segmentation (clustering)**

**remap** is the pre-existing NeuroScope module that performs interactive SOM segmentation using any of the SOM visualizations. A paired down functionality is also included in **NeuroGlimpse**.

**gsegSOM** (or Graph Segmentation of a Self-Organizing Map) is an integrated collection of modules which drives the *automated cluster inference* process from the prototypes of a learned SOM and generates summary outputs intended to guide and inform the human analyst. As such, it manages communication and converts input / output formats between several stand-alone, third party computational and visualization modules: NeuroScope modules **annCSOM, CONNvis, specter** (for neural manifold learning, CONN representation, and spectral cube and cluster examination), **igraph** [CN06] (for automated cluster capture) and CASA for scientific visualization and exploration of the spectral structure of inferred clusters. As a benefit to the human analyst, **gsegSOM** generates internal rankings and various summary visualizations (in standard, computer-ready formats such as .pdf and .png as well as .fits) of each clustering which facilitate their visual and statistical analysis. In various versions and forms, **gsegSOM** has been utilized for published results in the course of this Study ([MTI17, MIT18, TM19, MT19]).

Additionally, if a previous clustering for the given data already exists, the analyst can choose to compare it to **gsegSOM**'s clustering output via a process of reconciliation, which aligns (spatially) multiple clusterings, performs a statistical pixel-level matching of each, computes various (dis-)agreement measures relative to the (mis-)match, and ensures that cluster identification criteria (labels and colors) are aligned in accordance with this matching. This functionality has proved crucial to exercising **gsegSOM** in controlled settings for validation purposes.

The main output products of **gsegSOM** are (1) clustered images (in .png and .fits format), (2) clustered SOMs (in .pdf format), and (3) cluster-level statistical summary of the spectral signatures of each clustering (in .pdf format). For completeness, a typical data clustering experiment mandates multiple parameterizations of igraph clustering procedures (see paragraph on **DM-Prune**). To assess the comparative quality between different clusterings, (1) and (2) above are compiled into a "vistable"

which displays thumbnail snapshots of cluster footprints, across parameterizations. Visualizations of the organization of the SOM prototypes relative to each clustering are also produced; scrutinizing this visualization can help an analyst decide whether a particular clustering is under- or over-segmented, or violates certain paradigms of prototype-based clustering via CONN (i.e. severe non-contiguity of cluster footprints on the SOM grid).

The workflow of the **gsegSOM** procedure is roughly as follows:

1. Digest output of NeuroScope products
2. Parameterize the clustering routines
3. Perform automated clusterings via **igraph**, on a paremeter grid pre-set by the user
4. If applicable / desired, reconcile the clustering results to a known cluster structure (for each cluster, find the best-matching one in a template and inherit the template's color for the respective cluster)
5. Compute cluster-level statistical summaries (numerical and graphical) of each clustering
6. Visualize each clustering, on the SOM lattice (coloring each segmented group of SOM prototypes to a cluster color); and in data space (coloring each spatial pixel location to the color code of its SOM prototype)
7. Rank the clusterings by any of the available appropriate measures and produce a high-level "visual table" for comparison and recommendation
8. Generate .fits format output of each clustering, for further scientific analysis in a native .fits viewer (i.e., CASA)

The capabilities of the prototype **gsegSOM** package are documented in detail, in the usual R style, in Appendix4.3. We intend to turn **gsegSOM** into a regular R package (see section 2.6).

**DM–Prune** As previously mentioned, the automated graph-segmentation routines provided by the igraph package require multiple parameter selection: the main parameter required of each segmentation method is the adjacency matrix of SOM prototypes; secondary, algorithm-specific parameters are also required in some cases. Previous work segmenting SOM prototypes with the CONN matrix ([MT19]) has shown sparsification of the CONN graph can improve the quality of the captured clusters (as determined by their analysis, post clustering), with optimal quality achieved via a grid search of sparsity parameters. In attempt to automate this procedure, work has also been initiated under this grant to provide a framework for determining optimal graph sparsity *prior* to cluster capture (i.e., without the need to produce multiple clusterings from different sparse graphs and compare them). This process, dubbed **DM–Prune**, develops a Dirichlet-Multinomial model of the CONN edge weights and selects, via a Bayesian model selection approach, one optimal SOM adjacency which is then input to igraph. While this work is still new, results presented at conference (WSOM+ 2019, June 26–28, 2019, Barcelona, Spain, [TM19]) have shown **DM–Prune** capable of producing sparse CONN graphs which, when used as input to igraph algorithms, produce clusterings of comparable quality to those achieved by manual grid search procedures. Further experiments on **DM–Prune**d CONN graphs are needed to lend confidence in the method (and justify its inclusion to **gsegSOM**).

## 5. Components to perform evaluation of clustering quality

**specter** is designed for interactive viewing of spectral signatures from a selected spatial window of an image cube, in spatial context as an array of post stamp plots, and performing simple manipulations on the spectra within the plot array. The spatial window is selected from a reference image that represents the spatial dimensions of the image cube; for example, a continuum imge; a single channel map; or a

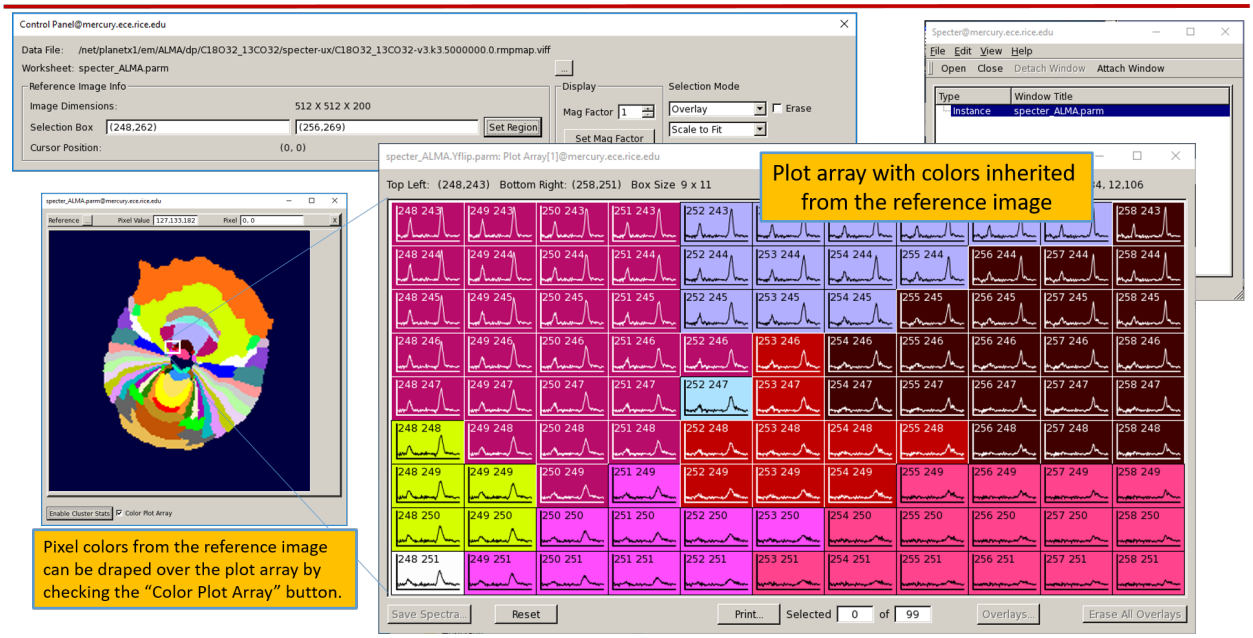
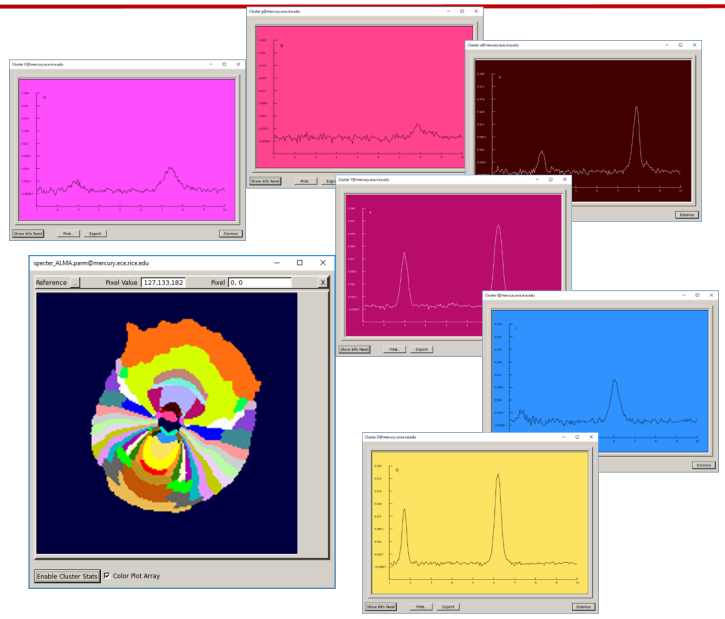



Figure 2: The user interface of **specter**, with a selection box shown in the reference image, and spectra within that box plotted in their spatial context in a plot array. Colors of the reference image pixels can optionally be transferred to the plot array. For better viewing, please see ppt slides in Appendix 4.1.

color cluster map. Examples of operations are overlaying one or more spectra on another spectrum by dragging/ dropping them; vieweing the spectra (including the group of overlain spectra) in large window by clicking on the post stamps; plotting the average of selected spectra; compare averages of different groups of pixels (take difference, ratio of the averages). Spectra in irregularly shaped spatial regions can be selected an manipulated. When the reference image is a cluster map (where colors signify different clusters), clicking on a cluster (a specific color) will bring up a plot of the mean cluster signature, with plot background color inherited from the cluster, and the letter label of the cluster shown. **specter** has capabilities to save plots in files; or to export a selected group of spectra into text files (one spectrum per file) encoding the pixel location in the file name. Details are given in the documentation of **specter**, section 4.7, and illustration in 2.5.

**specter** is in development. A Unix version exists with full functionality, however, the hosting Sparc stations have been discontinued, along with corresponding software libraries. Porting to Linux required a major change to different library base (Qt), due to which we are still discovering unexpected behavior occasionally when attempting to implement new features. (A current example is that implementation of pop-up of cluster mean plots (see illustration in Fig. 3 and section 4.1) caused incorrect labeling of the X axis.) We are dealing with such development issues on a continued basis.
Implementation of physically meaningful, intuitive coloring schemes for cluster maps was a study goal delayed by unplanned forced efforts (section 2.3). However, it is currently under development, and will be done in the next month or two. The ground work has been layed by the porting of **specter** to Linux / Qt library base. An intuitive scheme we plan - instead of randomly assigning colors to clusters - is to cast the frequency, width, and height of the intensity peak in each mean cluster spectrum to Hue, Saturation and Intensity color space. This will result in continuous change across the rainbow colors as the peak shifts, while (say) low to high saturation can indicate wide to sharp peaks, and brightness can



Cluster statistics are computed (as a one-time operation) when the Enable Cluster Stats button is pressed at lower left of the reference image. This assumes that the reference image is a cluster map.

Subsequently, clicking on any cluster (a distinct color) brings up the mean spectrum of that cluster (mean of all pixels of the same color), with the letter label printed at top left. The background of the plot inherits the color code of the respective cluster.

Plots can be printed to pdf files, and the numeric values can be exported to text files.

Figure 3: Mean spectra of clusters pop up in **specter** upon clicking on the clusters. For better viewing, please see ppt slides in Appendix 4.1.

express the level of intensity, all in one map.

An illustration of **specter**'s utility is in Fig. 2. Figs 3 and 4, and slides in 4.1 add to this illustration. Full user documentation is attached in section 4.7 to give a more complete sense of functionality.

**gsegSOM** and **specter** provide various ways to produce plots and/or text files with numerical output, for visual inspection, or for input to further manipulations in software of the astronomer's choice. For example, export spectra of selected pixels to individual ascii files; export cluster means to text files, save plots to pdf files, etc. Pre-existing modules **clstat** and **vecplot** have some overlapping funcionalities, as well as additional ones. **vecplot** can plot a set of spctra in a stacked fashion with intelligent calculation of the vertical offset spacing. We want to implement a similar choice in **specter** to display cluster mean spectra more compactly than in Fig. 3.
For assessing clustering quality in the absence of a template, several traditionally used as well as newer *Cluster validity indices* (CVI-s) are available in the NeuroScope module **cvi**, including our own CONN_index measure that is shown to judge complex cluster structure better than recent popular CVI-s [TM11]. Several graph-theoretical internal measures such as *modularity* can be used through **gsegSOM** from the **igraph** package (the call to **igraph** is tranparent to the user). **gsegSOM** also offers measures to compare a clustering to a template (external reference); Adjusted Rand Index, Jaccard Index, Normalized Mutual Information, Unweighted and Weighted Overall Accuray, and Cohen $\kappa$ (customary in remote sensing). We output lists, plots and/or "visual tables" with cluster map images in the order of their scores by selected quality measures.

**ClumpFind2DProj** and **ClusMap2FITS**, described above facilitate comparisons / examinations of clusterings across CASA and NeuroScope. We can either export a NeuroScope cluster map to CASA and examine it there including comparisons with ClumpFind clusters; or use **specter** in NeuroScope with a 2D "projection" of ClumpFind clusters (produced by **ClumpFind2DProj**) as the reference

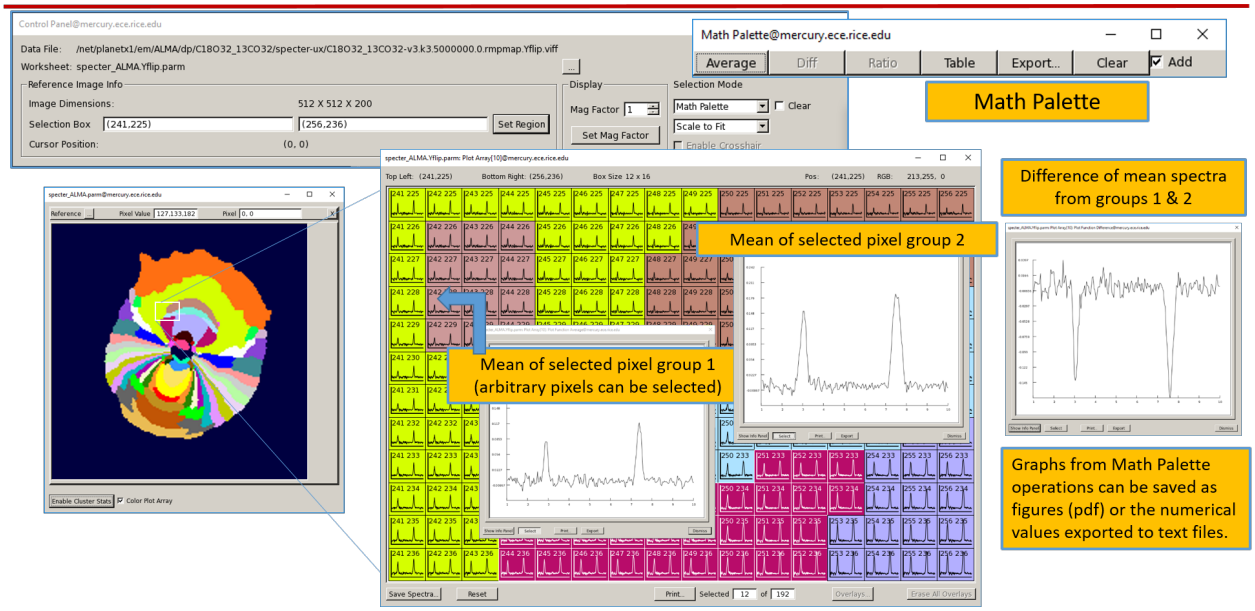



Figure 4: Math palette function in **specter**. For better viewing, please see ppt slides in Appendix 4.1.

image (see in Fig 2).

## 2.5 Demonstration of NeuroScope capabilities on ALMA data

The sample analysis results discussed in this section were produced by the tools described in section 2.4. Tool/software module names are resolved in Fig. 1 and further detailed in section 2.4. Sample workflows are shown in attached slides, section 4.1. This section shows results produced by our tools on real and synthetic ALMA data. Testing of NeuroScope capabilities has also been done on other data, importantly, synthetic multi-, hyperspectral image cubes where ground truth is available for all pixels; and on terrestrial remote sensing imagery where detailed field knowledge and comparison to previous classification / clustering could serve for quality assessment. These studies have been published in [MT17, MT19, TM19], papers attached in section 4.8.
The following results demonstrate that our SOM learning and automated clustering based on the SOM-derived CONN graph a) achieve the level of detail and cluster discrimination comparable to interactive clustering by human expert; b) extract almost as much structure from the weaker lines (e.g. $C^{18}O$ lines) as from stronger lines (e.g., $^{13}CO$ lines); and c) the combination of the two lines yields potentially interesting information not present in the single-line clusterings. We also show automatic ranking of clusterings produced on a parameter grid, to recommend top-ranked ones for closer examination by the astronomer.

**Analysis of single-line vs multiple-line cubes of protoplanetary disk HD 142527.** The right panels of Fig. 5 shows clustering maps of the ALMA observations of the HD 142527 protoplanetary [BWI+17] produced by running Neuroscope on both single-line data cubes (C18O32 and 13CO32) and on the combination of both data cubes using interacting (I-C) and automated clustering (C18O32_13CO32). The main characteristic of all the clustering maps is the identification of morphological features (i.e.

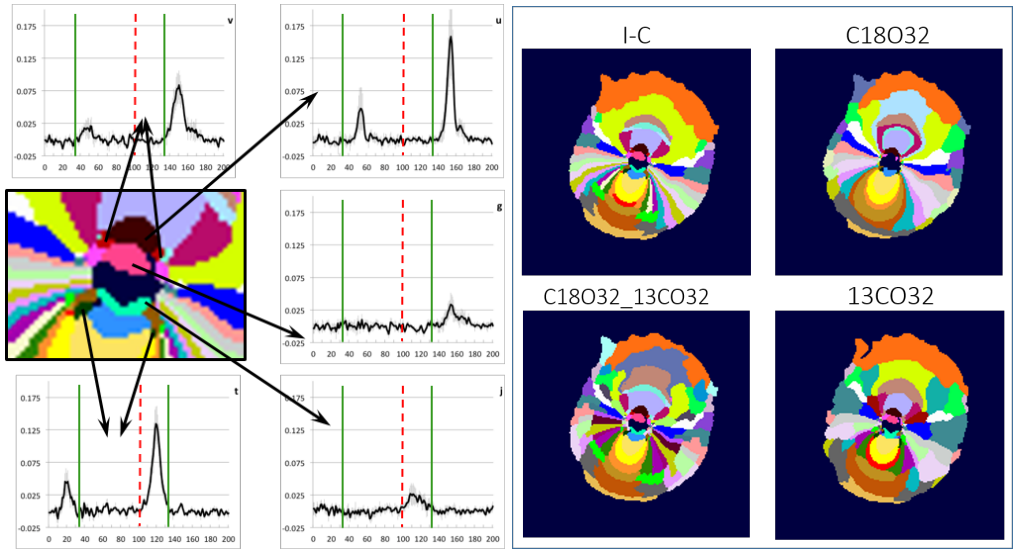



Figure 5: Cluster detail obtained from single-line vs combined-line ALMA cubes. Left panel: the center of the interactively clustered HD 142527 disk, shown fully at upper left in the right panel. Arrows from several small clusters and their mean spectral signatures are pointed out. This clustering was done from stacked cubes of $C^{18}O$ and $^{13}CO$ lines, each comprising 100 channels. The red vertical dashed line indicates where the two lines were concatenated. The green lines are drawn at the rest frequency. Right panel: Interactive clustering by human expert (I-C), top left, compared to automated clusterings from single $C^{18}O$ and $^{13}CO$ lines, in right column, and to automated clustering from the combined lines, bottom left.

the shape of the clusters) that correspond to the iso-velocity lines of a rotating disk. The shape of these clusters closely resemble the maps of the first moment of the intensity shown in Fig. 6. However, the joint analysis of the two emission lines reveal small clusters that indicate deviations from Keplerian rotation. For example, the left panel of the figure shows that the center of the I-C clustering map is characterized by several small clusters with potentially interesting kinematics (the arrows point to the mean spectra of the clusters). In particular, as discussed in [MTI16], the spectrum of cluster "t" has two peaks in the $^{13}CO$ line on opposite sides of the rest frequency indicated by the green vertical lines.This hints to two gas components moving in opposite directions, possibly due to vertical motions symmetric with respect to the disk midplane.
The right column of the right panel displays the top-ranked automated clusterings from the single lines, while the best automated clustering from the combined lines is at bottom left of the right panel. First, the automated clusterings produce comparable structure and detail to that in the interactive clustering (I-C). Second, there are strong similarities, as well as systematic differences:

- The single-line cluster maps miss different subsets of the small clusters seen in the magnified center of the I-C clustering.
- The multi-line automated C18O32_13CO32 clustering detected the small clusters. The top ten or so of the multi-line clusterings have all small clusters. (See the top 3 choices by various measures in 2.5 and sample table with ranked cluster maps in 4.4.)
- In the 13CO32 and C18O32_13CO32 maps an inner dial is outlined at the distance of the jog(s) present in the I-C clustering.

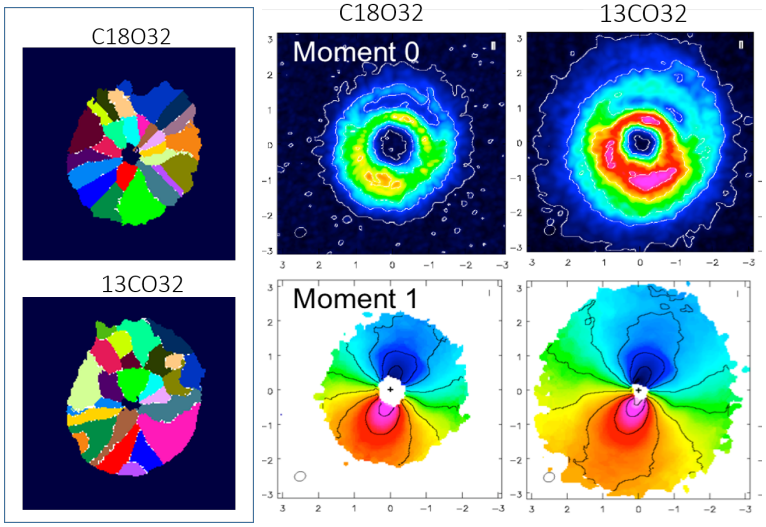



Figure 6: Comparison of NeuroScope clusters of HD 142527 disk in Fig. 5 with ClumpFind clusters for the $C^{18}O$ and $^{13}CO$ (3-2) lines (top and bottom-left, respectively), and with maps of the zeroth (top center and right) and first (bottom center and right) moments of the line intensity. All the panels have the same angular size, though the emission have different extension due to different threshold levels used to remove pixels affected by noise.

- In the C18O32 cluster map the same radial features show as in the I-C clustering, with the same jogs.
- Notably, significant structural details emerge from the data of the weaker C18O32 line.

Since we do not have labeled data ("template") for the evaluation of the results on real ALMA objects, we compare the cluster structure obtained by NeuroScope with independent analyses of the same data based on maps of the intensity moments and clusters generated by using the CUPID package of clump-finding algorithms [BRJE07]. The latter provides a set of tools for identifying and analysing clumps of emission within 1, 2 or 3D data arrays allowing the user to chose between different clustering algorithms (see documentation at http://starlink.eao.hawaii.edu/docs/sun255.htx/sun255.html). In the following we adopted the ClumpFind algorithm which, quoting from CUPID manual "contours the data array at many different levels, starting at a value close to the peak value in the array and working down to a specified minimum contour level. At each contour level, all contiguous areas of pixels that are above the contour level are found and considered in turn. If such a set of pixels includes no pixels that have already been assigned to a clump (i.e. have already been identified at a higher contour level), then the set is marked as a new clump. If the set includes some pixels that have already been assigned to a clump, then, if all such pixels belong to the same clump, that clump is extended to include all the pixels in the set. If the set includes pixels that have already been assigned to two or more clumps, then the new pixels in the set are shared out between the two or more existing clumps. This sharing is done by assigning each new pixel to the closest clump. This process continues down to the lowest specified contour level, except that new clumps found at the lowest contour level are ignored." Generally, we ran the ClumpFind algorithm with changes made to the minimum spatial size of clumps and the spacing in between the contours used to evaluate clumps. These clump-finding algorithms work only with intensity and proximity; they do not account for the kinematics, geometry, or turbulence of a system. This is appropriate for finding dense regions of molecular gas on a parsec or sub-parsec scale, but is not useful

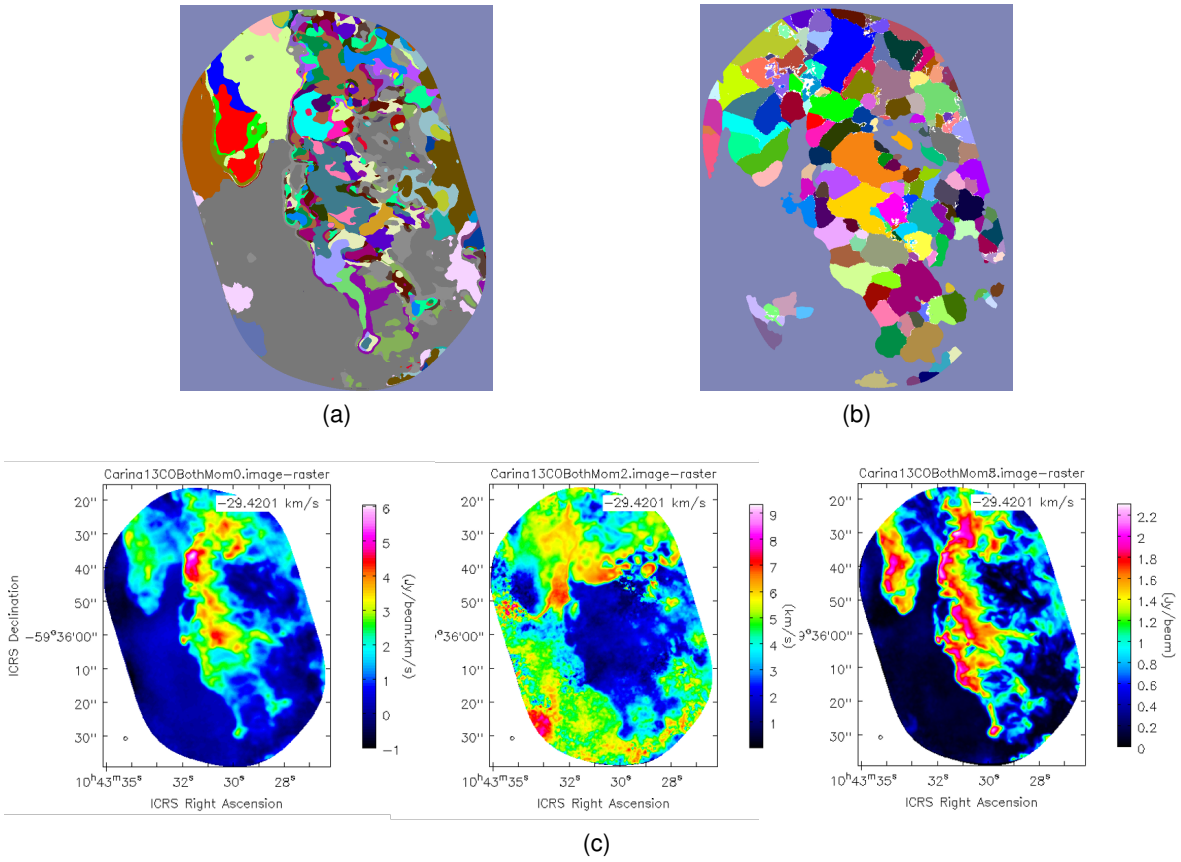



Figure 7: Comparison of cluster detail obtained from $^{13}$CO line ALMA cube, which contains two distinct components in the Carina Nebula each manifesting in a separate frequency zone within the line: the Western Wall, channels 40 - 100, and the "Northern Region", channels 120- 165. (a) NeuroScope clustering from combined channels 40-100 + 120-165. (b) ClumpFind clusters from the combined channels, projected to 2 dimensions with our **`ClumpFind2DProj`** module. (c) Moment maps 0, 1, and 8 (peak intensity map) from the combined channels.

for probing the structure of a rotating disk. Our disk analyses done with ClumpFind are thus far inferior to those done with Neuroscope.

The left panel in Fig. 6 illustrates that ClumpFind is not well-suited for finding clusters in protoplanetary disks where the dominant structure is that of regularly rotating material. Instead, NeauroScope clusterings shown in Fig. 5 bear clear resemblance to the first moment maps in the right panel of Fig. 6 but also carry information about the velocity integrated intensity (zeroth moment), as can be seen by comparing the shape on the clusters in the top part of the image to those in the bottom. This underlines the relevance of clustering approaches like NeuroScope that make no prior assumption about the nature of the spatial structure to be discovered. The ppt file *`Analysis-summary.ALMA.pptx`* in Appendix 4.2 contains more comparisons including top-ranked automated clusterings by several quality measures.

**Analysis of Carina Nebula.** As a second example, we discuss NeuroScope clustering analysis of $^{13}$CO and C$^{18}$O observations of a PDR region in the Carina Nebula obtained as part of project 2015.1.00656

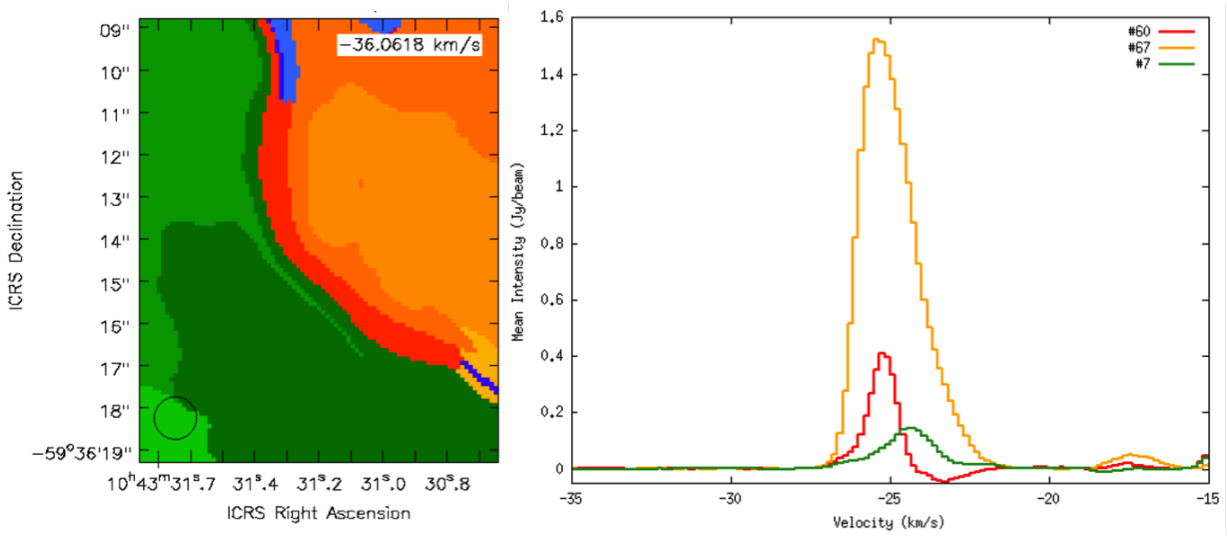



Figure 8: Left: CASA Viewer zoom-in of the NeuroScope clustering map of the $^{13}$CO (2-1) emission recorded toward Carina. Right: Mean intensity spectra of cluster 60 (red line/cluster), 67 (orange line/cluster), and 7 (dark green line/cluster). The color of the clusters visualized by the CASA Viewer are not reconciled to those used by NeuroScope shown in Fig. 7.

(see 2.3). This region is characterized by a complex gas morphology and kinematics as illustrated by the moments maps shown in panel c of Fig. 7. Manual inspection of moment maps reveal the presence of multiple structures with shapes varying from stretched filaments to round clumps. However, moment maps do not provide any clear indication of coherent structures and provide little help in understanding the physical properties of the observed region.
On the other hand, NeuroScope and ClumpFind produce very different cluster maps (panel a and b, respectively). While the outer boundary of the emission is similar (this is controlled by the noise level), Neuroscope seems to be very successful in finding clusters that correspond to coherent kinematic features, whereas ClumpFind seems to group nearby pixels in clusters of similar size regardless of their intensity or kinematics. For example, NeuroScope recognizes and groups all the pixels located at the interface between the molecular cloud and ionizing radiation front in elongated filamentary structures that share similar intensity and velocity dispersion (Fig. 8). This is a very important result since the scientific goal of the project was to investigate the distribution and kinematic of molecular gas behind the PDR.

The ppt file *Analysis-summary.ALMA.pptx* in Appendix 4.2 contains comparisons focusing on frequency sub-regions (in the $^{13}$CO line ALMA cube) of the Western Wall and Northern Rgeion separately. This allows more direct evaluation of NeuroScope vs ClumpFind clusters.

**Analysis of synthetic disks.** In order to test NeuroScope capability of discovering coherent structures in multi-dimensional data, as well as to support the astronomical interpretation of clustering maps, we performed the analysis of well-controlled models of the CO line emission of circumstellar disks. These models were characterized by fixed physical properties (e.g., surface density, gas temperature, mass of the stellar mass) but were observed at different viewing angles. This led to different morphologies of the line emission. The CO emission was produced using the radiative transfer code RADMC3D, while

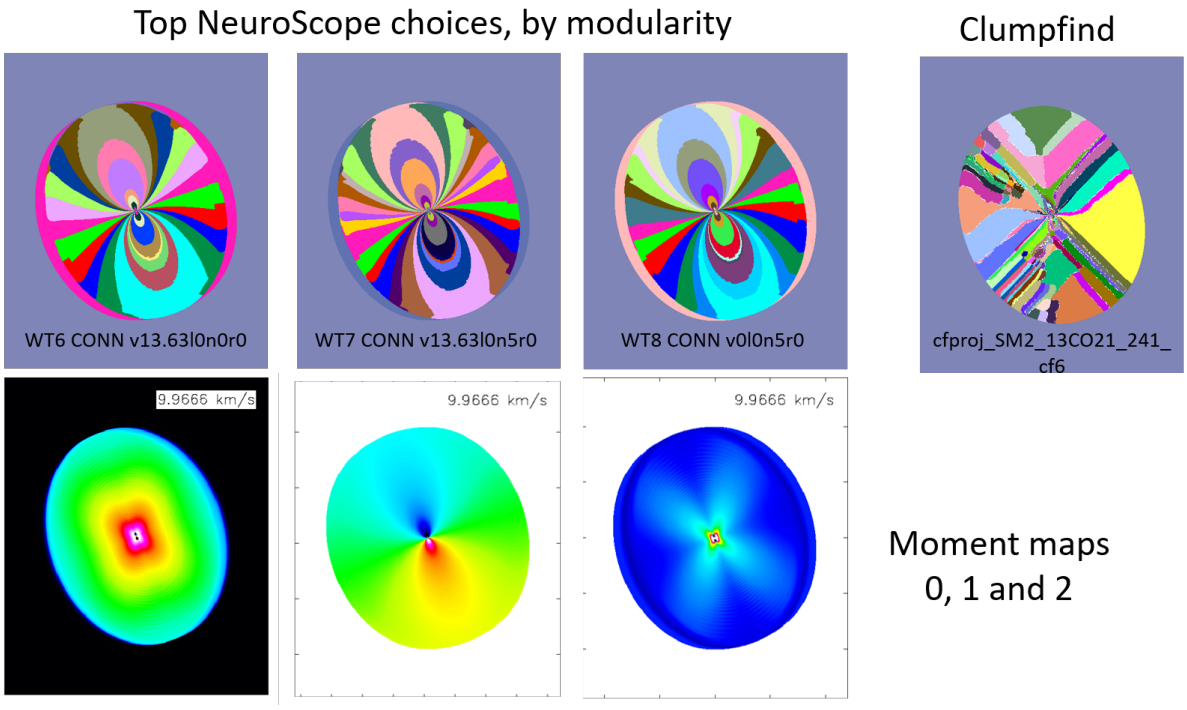



Figure 9: The best three automated clusterings of a synthetic disk (left panel in top row), compared with ClumpFind clustering (top right) and with moment maps (bottom row).

synthetic ALMA data cubes were generated using CASA task "simobserve" which accounts both for discrete uv-sampling and thermal noise. The distribution of gas in the disk models is center-symmetric while the spatial distribution of the line emission is controlled by the interplay between the Keplerian rotation of the disk and the viewing angle, which, for viewing angles different from face-on, leads to the well known Doppler shift pattern of Keplerian disk. Figure 9 illustrates the case of a Keplerian disk inclined by 40° and rotated by 110° counterclockwise. The bottom panels show the zeroth, first, and second intensity moments (integrated intensity, velocity centroid, and velocity dispersion, respectively), while the top panels show clustering maps obtained with NeuroScope and ClumpFind. The analysis the same disk model observed face-on (zero inclination) is presented in Analysis-summary.ALMA.pptx in the Appendix. The NeuroScope clustering maps shown in Figure 9 are the best three automated clustering maps and slightly differ in the number of clusters. However, the shapes of the clusters are very similar and trace the iso-velocity curves of an inclined rotating disk as shown by the first moment map. At opposite, ClumpFind fails in grouping pixels based on their kinematic properties and returns a clustering map with not physical meaning. In addition of properly grouping pixels with similar spectral properties, NeuroScope identifies crescent-shaped clusters on both sides of the disk emission which correspond to lines of sight (pixels) that intersect the vertical edge of the disk at the disk outer boundary. Similar features are visible in the zeroth and second moment maps.

Despite the fact that the line intensity is symmetric with respect to the projected minor axis of the disk, the NeuroScope cluster structure is slightly asymmetric meaning that the clusters in the top half of the disk are not the exact mirror of those in the bottom half. This effect is unexpected and might be caused by the fact that while the intensity is symmetric with respect to the disk minor axis, the velocity at which the line is emitted, and therefore the spectrum of each pixel, is symmetric with respect of the

systemic velocity of the system, which in this case has been assumed to be zero. The effect is small and probably does not interfere significantly with cluster identification, but we continue to investigate the origin of it. It seems to be specific to this synthetic spectral cube in that it occurs with the tilting of the disk, i.e., when the Doppler shift of the line emission due to the disk rotation is introduced. We have not seen similar effect on other synthetic data such as spectral image cubes of non-symmetric objects (like urban scenes in [MT19, TM19]).

## 2.6 Path to software deployment to ALMA

In this section we elaborate the challenges and requirements involved in transferring various NeuroScope components to the ALMA environment.

**gsegSOM** is an R package designed to automate the clustering of SOM prototypes and produce cluster maps (images) and associated cluster statistics. Original package development occurred in macOS Sierra (10.12.5) but it has also been deployed to Red Hat Enterprise Linux Workstation release 7.6. Package sub-modules are written in both R and C++11 and have extensive dependencies upon libraries native to both languages. Notable C++ dependencies include the Armadillo C++ library for linear algebra (http://arma.sourceforge.net/, which further requires FLIBS — a collection of Fortran modules — http://flibs.sourceforge.net/ and LAPACK and BLAS routines) and the ImageMagick library for image manipulation (https://imagemagick.org/). Most (but not all) R package dependencies are available via the official CRAN repository for R (https://cran.r-project.org/).
Making **gsegSOM** available for public use requires a close alignment of system configurations, as we discovered upon migrating from macOS to Red Hat Linux, but system dependency checks have not been suitably developed to streamline this process (since we work locally, we achieved system congruency via trial and error, but that is not feasible for widespread distribution). Additionally, submitting packages to the CRAN repository (the proper way to publicly distribute an R package) demands considerable effort (a good summary of this process can be found at http://r-pkgs.had.co.nz/release.html). The most relevant challenges this process imposes for **gsegSOM** are:

1. CRAN requires an active package maintainer; packages without active maintenance are systematically removed from the repository. This would require someone with continuing obligation to provide package maintenance as requested by CRAN.

2. CRAN requires package releases to not disrupt any downstream dependencies (i.e., to other packages which depend upon it); this poses no impediment upon a primarily release of **gsegSOM**, but if others were to develop tools based upon it the functionality could become constrained.

3. Cross-platform development is required; we have successfully ported the package from macOS to Linux, but Windows development is missing.

4. CRAN dissuades package authors from providing functionality that modifies a users local file system without explicit consent; this includes writing local files, which **gsegSOM** does liberally. We are unclear whether explicit consent is required at each instance of file writing, or can be given to an entire work session. Regardless, it would be CRANs decision as to whether this crucial functionality violates their policies so guidance from their team would be required, which might result in structural modification to **gsegSOM** workflows.

Deployment of **DM-Prune** involves similar issues as **gsegSOM**.

**NeuroGlimpse** is written in MATLAB; it should be relatively easy to deploy. However, differences in MATLAB environments on different platforms can cause problems. We experienced this between Linux red Hat and Windows. Similarly to R packages, stable solution would be provided by making it a formal Toolbox, or at least testing it on several popular platforms. Either of these is currently beyond our resources. We provide the code "as is". We request to restrict its use to a few volunteer testers (identified by us or by NRAO) until we had a chance to publish it. We will interact with testers to provide help in return for feedback.

**specter** was written over a number of years in unix environment by NASA Space Grant students under the mentorship, and by the design of the PI. A subsequent update was started around 2010 to use Qt libraries instead of the outdated unix xwindows and Olit functions, in part as a rescue from an environment no longer supported, and in part as preparation for easy porting to multiple platforms in order to facilitate technology transfer. Update efforts succeeded to put **specter** on Qt basis in Lunix environment at that time, but fell short (for reasons of DARPA funding cuts) in transferring many of the scientific functionalities. In this Study, our undergraduate student has completed the porting of most "missing" functionalities in the Qt-updated version. However, **specter** has dependencies on image and data cube manipulation libraries of **khoros** [RY92], a signal and image processing package. Transferring **khoros** to modern platforms would require recompiling the libraries for 64-bit machines, but we do not have source code. (Software distribution of **khoros** did not include source code.) Two possible avenues are 1) replace the **khoros**-supported functionalities by other equivalent third-party software if such one exists, or re-write the respective functionalities by ourselves; or 2) try to purchase the source code. Both require more resources than we can manage from small or medium-size grants.

**NeuroScope – CASA interface** for import / export of products we developed in this Study would be easy to integrate.

**NeuroScope** in its entirety (except for the R packages): Core modules (**annCSOM, CONNvis, TopoView, remap, specter** in Fig. 1, and more), with support features for development, housekeeping, testing, interfacing with other modules, sophisticated data handling for data cubes and meta data, extensive data structures for "brains"; GUI-s, etc. capitalize on one or more of three major third-party libraries. One is Qt, freely available and easy to install.
The second one is **khoros**, described above, therefore transfer of a number of modules would involve replacement of **khoros** functions or obtaining the source code to recompile for current Linux (or Windows) 64-bit architectures. These functions work perfectly, in fact with some non-trivial system library substitutions we even managed, by trial and error, to compile the flagship module **annCSOM** (using the run-time **khoros** libraries) on a 64-bit Linux machine; however, the **khoros** functions themselves had been compiled for 32-bit architecture, which severely limits their address space. (For example, because of this limitation **annCSOM** cannot process image cubes larger than approx. 2Gbytes a piece, no matter how much memory we have.) The path to remedy this is as described above: either re-write and replace **khoros** functions, or obtain the source code and properly recompile for targeted architectures.
The third major third-party library **NeuroScope** depends on is Neural Works Professional Plus [Neu03], by NeuralWare, Inc., http://neuralware.com/. The PI's group have been utilizing NeuralWare's deployment package ("dpack", purchased as source library) to interface with extensive data structures Neural Works Professional Plus provides for storing "brains" (neural networks) and with standard learning and other functions to use networks deployed from the GUI-driven Neural Works Professional Plus package. This allowed focusing (small-to-medium size) grant resources on building unique NeuroScope capabilities on top of this library base avoiding the coding of many basic functionalities. Distribution of deployed networks and code we write to manipulate them (i.e., NeuroScope) is not restriced by NuralWare license.

Integrating NeuralWare libraries should be a matter of purchasing licenses and installing software.

The **RAPTOR** parallel FPGA board for acceleration of SOM learning (decribed in section 2.4) has standard PCI interface; and is driven by its own MATLAB-based GUI, its operation is fairly standard. Installing it in the ALMA pipeline would mostly be a matter of purchase and maintenance / support issues.

### 2.7 Budget status

Funds have been exhausted without overspending. Final invoicing has been initiated. Owing to delays caused by data issues outside of our control (as described in section 2.3) we had to request, and were granted, a 9-month no-cost extension. Funds originally budgeted for one year were stretched by cutting PI and Co-PI summer salaries and reallocating to cover graduate student support.

### 2.8 Publications and presentations

**Refereed journal and conference papers.** Papers produced during this Study are listed in 4.8 and are attached as Appendices.

Most papers are also downloadable at http://www.ece.rice.edu/∼erzsebet/publist-Merenyi.pdf

**Presentations:**

- Discovery from Hyperspectral ALMA Imagery with NeuroScope. Merényi, A. Isella, Taylor, J., ALMA 2030 special session, USNC-URSI National Radio Science Meeting, Boulder, CO. January 5, 2018
- Neural Machine Learning for Discovery and Interpretation in Complex ALMA Data. Merényi, A. Isella, Taylor, J. National Radio Astronomy Observatory, Charlottesville, VA, November 17, 2017
- SOM-empowered Graph Segmentation for Fast Automatic Clustering of Large and Complex Data, Mernyi, E., Taylor, J., Proc. 12th International Workshop on Self-Organizing Maps, WSOM+ 2017, Nancy, France, June 27-29.
- A Probabilistic Method for Pruning CADJ Graphs with Applications to SOM Clustering. Taylor, J., and Mernyi, E., Proc. 13th International Workshop on Self-Organizing Maps, WSOM+ 2019, Barcelona, Spain, June 26-28, 2019.
- Dense Cores in the Chaotic Carina Nebula, M. Hummel, J. Taylor, E. Merényi, A. Isella, P. Hartigan, poster presented at Radio/Millimeter Astrophysical Frontiers in the Next Decade host in Charlottesville , VA. June 25, 2019.

## 3 References

## References

[BRJE07] D.S. Berry, K. Reinhold, T. Jennes, and F. Economou, *CUPID: A clump identification and analysis package*, Astronomical Data Analysis Software and Systems XVI ASP Conference Series **376** (2007), 425.

[BWI+17] Y. Boehler, E. Weaver, A. Isella, L. Ricci, C. Grady, J. Carpenter, and L. Perez, *A Close-up View of the Young Circumbinary Disk HD 142527*, ApJ **840** (2017May), no. 1, 60, available at 1704.00787.

[CN06] Gabor Csardi and Tamas Nepusz, *The igraph software package for complex network research*, InterJournal **Complex Systems** (2006), 1695.

[HIL+18] Pinghui Huang, Andrea Isella, Hui Li, Shengtai Li, and Jianghui Ji, *Identifying Anticyclonic Vortex Features Produced by the Rossby Wave Instability in Protoplanetary Disks*, ApJ **867** (2018Nov), no. 1, 3, available at 1809.07001.

[HTM+19] M. Hummel, J. Taylor, E. Merényi, A. Isella, and P. Hartigan, *Dense cores in the chaotic carina nebula, poster presented at ngvla19 conference*, 2019.

[LMPR13] J. Lachmair, E. Merényi, M. Porrmann, and U. Rückert, *A reconfigurable neuroprocessor for self-organizing feature maps*, Neurocomputing **112** (2013), 189–199.

[MIT18] E. Merényi, A. Isella, and L. Taylor, *Discovery from hyperspectral alma imagery with neuroscope*, 2018 united states national committee of ursi national radio science meeting (usnc-ursi nrsm), 2018Jan, pp. 1–2.

[MT17] E. Merényi and J. Taylor, *SOM-empowered graph segmentation for fast automatic clustering of large and complex data*, 12th international workshop on self-organizing maps and learning vector quantization, clustering and data visualization (wsom+ 2017), 2017June, pp. 1–9.

[MT19] Erzsébet Merényi and Joshua Taylor, *Empowering graph segmentation methods with soms and conn similarity for clustering large and complex data*, Neural Computing and Applications (2019June).

[MTI16] E. Merényi, J. Taylor, and A. Isella, *Mining complex hyperspectral ALMA cubes for structure with neural machine learning*, 2016 ieee symposium series on computational intelligence (ssci), 2016Dec, pp. 1–9.

[MTI17] Erzsébet Merényi, Joshua Taylor, and Andrea Isella, *Deep data: discovery and visualization. application to hyperspectral ALMA imagery*, Proceedings of the International Astronomical Union **12** (2017), no. S325, 281–290.

[MTZ09] E. Merényi, K. Taşdemir, and L. Zhang, *Learning highly structured manifolds: harnessing the power of SOMs*, Similarity based clustering, 2009, pp. 138–168.

[Neu03] NeuralWare, *Neural computing, neuralworks professional ii/plus*, 2003.

[RY92] J. Rasure and M. Young, *An open environment for image processing software development*, Proceedings of the spie/is&t symposium in electronic imaging, 1992February 14.

[TM09] K. Taşdemir and E. Merényi, *Exploiting data topology in visualization and clustering of Self-Organizing Maps*, IEEE Trans. on Neural Networks **20** (2009), no. 4, 549–562.

[TM11] ———, *A validity index for prototype based clustering of data sets with complex structures*, IEEE Trans. Systems, Man and Cybernetics, Part B **41** (2011August), no. 4, 1039–1053. DOI: 10.1109/TSMCB.2010.2104319.

[TM19] J. Taylor and E. Merényi, *A probabilistic method for pruning CADJ graphs with applications to SOM clustering*, 13th international workshop on self-organizing maps and learning vector quantization, clustering and data visualization (wsom+ 2019), 2019June, pp. 44–54.

# 4 APPENDICES

The following are attached to this Report as separate files.

### 4.1 Power Point slides summarizing NeuroScope capabilities and deployment path to ALMA, file *NeuroScope Tools and Workflows.pptx*

### 4.2 Power Point slides with clustering illustrations, file *Analysis-summary.ALMA.pptx*

### 4.3 Documentation for `gsegSOM`

### 4.4 Sample output products from `gsegSOM`

### 4.5 `NeuroGlimpse` Matlab code

### 4.6 Documentation of full NeuroScope SOM learning capabilities (module `annCSOM`)

### 4.7 Documentation for `specter`

### 4.8 Papers produced during this Study

[MT17, MTI17, MIT18, MT19, TM19, HTM+19] from References, section 3

In addition, currently in press:
Giuseppe Longo, Erzsébet Merényi, Peter Tiňo, Foreword to the Focus Issue on Machine Intelligence in Astronomy and Astrophysics. (editorial). *Publications of the Astronomical Society of the Pacific*. In press. 2019.