

Astronomical Data Visualization Tools

Bryce Billing
University of Cape Town

ABSTRACT

With the increasing need to develop good astronomical data visualization tools due to the creation of large radio telescopes such as SKA (Square Kilometer Array) the analysis of the current tools available will allow for a better grasp of what is lacking in the current market and what is needed to fulfill the requirements of such tools. This review gives an overview of the current astronomical visualization tools namely 3D Slicer, Karma, GIPSY, Euro3D, AstroVis, CASA, DS9, VisIVO, FITS3D, FITS2D and CyberSKA Viewer. The lack of uniformity in the data formats of these tools is a definite problem currently faced by the visualization tools available as well as the fact that there is no single data tool that meets the goals of all the users available. The existing tools although powerful in their own way do not currently meet the demands of the users and therefore need to be evaluated to see what can be done to improve this situation.

CCS Concepts

• Computer Science → Visualization

Keywords

Visualization, Visualizing Data Cubes, Visualization Tools.

1. INTRODUCTION

Visualization of astronomical data has never been as important as it is today due to the large amounts and greater diversity of data, per astronomer, than ever [20]. Visualization has a critical role in maximising the return from astronomical data in the current Era of Petascale Astronomy [15]. The aims of visualization of astronomical data is to allow for highly accurate and precise quantitative analysis as well as produce visuals to transmit the results beyond the scientific community [17]. We discuss a range of current tools available for visualizing astronomical data cubes. We do not give a highly in depth analysis of each of the tools, just specific information such as, who developed them, the purpose of the application, file formats, features, strengths and weaknesses, whether it supports distributed or local data processing and what they focus on i.e. 2D or 3D astronomical visualization, allowing for a comparison between the available tools. These aspects of the tools were specifically chosen so we can determine what aspects to keep when developing future visualization tools. This is important because in order to be able to create a good visualization tool in the future, the existing tools need to be analyzed to see what can be improved and what features should be kept. Analysis of the visualization tools also helps identify how certain challenges such as, the non-uniformity of data formats and visualizing the large amounts of data being produced by the large radio telescopes such as Square Kilometer Array (SKA) and MeerKAT, are handled by other tools and how these methods can be improved and incorporated in the design of future visualization tools. Most of the visualization tools render slices of a data cube: a 3D volume comprising x, y and frequency. Visualizations range from maps of spectral lines through separate slices of data at multiple wave-lengths to 3D contours [13]. An image of a data cube is shown below in Figure 1 below.

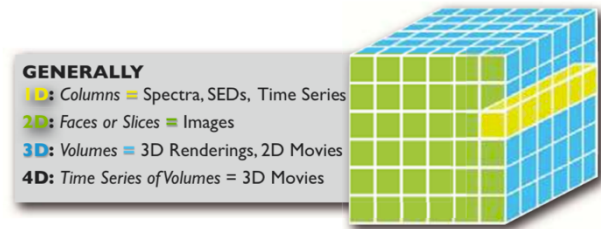


Figure 1. Data Cube [13]

2. EXISTING VISUALIZATION TOOLS

There are a number of existing visualization tools available for the visualization of astronomical data. The following tools are a select few of the tools available and are analyzed in order of most widely used in radio astronomy according to Dr. Sarah Blyth [24].

2.1 Karma

Karma is a visualization library that was developed and used by Commonwealth Scientific and Industrial Research Organisation (CSIRO) in Australia to develop visualization tools [1]. It was designed to provide a toolkit to help with developing visualization applications as well as a suitable environment for production tools. Karma supports multiple file formats: FITS, Miriad n-dimensional array, PPM and Sun Rasterfile [1] and due to its age it has extensive documentation. Karma supports both 2D and 3D visualizations.

Karma has a variety of features namely an integrated communication facility and an integrated portable graphics library to enable support for visualization applications [1]. Another one of Karma's features is functional concepts, which are grouped into packages and these packages provide both class and member functions [1]. The library is layered and each layer contains one or more packages. Packages at higher levels provide more power and packages also have no dependencies on other packages on any level. The level is also determined by the packages name. These specific features are added to implement the design goal stating that, it should be obvious as to which system provides the greatest flexibility, power and ease of use [1].

Karma's strengths include, allowing for high usability as it does not force the user to do things the "Karma Way" [1]. The key design goal of Karma is to provide a simple way to navigate to allow the library to be easily used [1]. Karma's age is also a strength because it means that it has had a lot longer to improve and develop features required by the community.

However, there are problems with Karma, it being old software is also a weakness because the data volumes being produced now by facilities such as SKA are much larger than facilities at the time of Karma's development. Mohan et al. (2016) mentions that a weakness of Karma is that it is slow, this is probably due to the increase in the volumes of data being produced as mentioned earlier. Mohan et al. (2016) also states that tools like Karma cannot generally be used easily on a commodity laptop.

2.2 GIPSY (Groningen Image Processing System)

GIPSY was developed at the Kapteyn Astronomical Institute at the University of Groningen, to help visualize multi-dimensional data [9]. GIPSY was developed because complex data being produced was not being used to its full potential by the astronomical community, yet it contains a large amount of information [9]. GIPSY supports the GIPSY file format [19]. It has poor documentation and its main visualization mode is 3D. Despite GIPSY having poor documentation, it does have information on a list of features [9]. These features include, powerful analysis and galaxy modelling which is a feature and a strength. GIPSY supports a large variety of coordinate systems [11] and GIPSY's core functionality includes support for applications for modelling rotation curves, interactive inspection and characterization of 3D data [9]. GIPSY also supports local data processing.

GIPSY is a mature and powerful visualization tool and its core functionality needs to be maintained when implementing improvements for GIPSY, as these features are not very common in the current visualization tools available [9].

GIPSY does have some problems, in that it needs improvement of its Graphical User Interface (GUI), better and more detailed documentation as well as it needs to have an easily installable and maintainable system [9].

2.3 CASA (Common Astronomy Software Application) Viewer

CASA is an image viewer that was developed by the National Radio Astronomy Observatory in the United States of America, to meet the requirements of the ALMA and EVLA telescopes as well as to meet the intention of not just meeting those requirements but build a general purpose data analysis tool for radio, millimeter and submillimeter telescopes [8]. It supports a variety of file formats and data types: CASA, FITS, ALMA, EVLA/VLA and UVFITS [8]. CASA does have extensive documentation and is primarily a 2D visualization tool.

CASA's features include, local data processing, supporting operations such as data flagging, playing animations by stepping through the z-axis and the overlaying of images. It also supports non-interactive and interactive clean modes during imaging, the difference between the two being that in the interactive mode the user can make selections during the cleaning process while in the non-interactive mode the user cannot [8]. CASA allows the user to create multiple interactive plots at once, as well as it being able to display multiple graphical user interfaces. More features are that it can support a large number of frequency channels and a spectral profile can be plotted in a separate window [8]. The functionality of CASA is also separated into two categories, namely tasks and tools [7].

CASA is powerful [8], tools provide flexibility and functionality [8] as well as allow access to lower level features [7]. Tasks are more user friendly [8] and provide higher level interface to frequently used applications [7]. Another of CASA's strength is that it allows for multiple interactive plots and GUI's [8].

Some of the problems with CASA's visualization is that it does not support the entire region shapes and parameters and the viewer may not properly open saved region files [21].

2.4 SAOImage DS9

DS9 was developed by the Smithsonian Astrophysical Observatory in the the United States of America. It was developed to view FITS images and binary tables [1]. DS9 supports the FITS file format and is thoroughly documented [10]. DS9 is a 3D visualization tool and supports many features, namely FTP and HTTP access which allows for easy communication with other tools, remote analysis, support for the Virtual Observatory therefore allowing easy communication with other tools, a built in help facility, web based archive servers and is able to bin fits event files into a 3D data cube [10]. It also has support for FITS multiple WCS and allows the user to toggle the display of various coordinate systems and has support for composite regions and templates as well as an overlay display [11]. DS9 has full support for loading, displaying, filtering and saving catalogues (a secondary parameter defined from processing images [15]) [11]. Region annotation with text and simple shapes is also supported by DS9 [3].

DS9's strengths are its simple GUI and simple installation process [22]. DS9's weaknesses are that it is a slow application when processing large amounts of data and it also cannot be used on a commodity laptop [3].

2.5 3D Slicer

The Astronomical Medicine (AstroMed) Project at Harvard University's Initiative in Innovative computing decided to use medical imaging and visualization tools such as 3D Slicer to visualize astronomical data [5]. 3D Slicer was originally developed at the MIT Artificial Intelligence Laboratory [5]. It was first used as a 3D brain imaging tool and was used to help surgeons in image guided surgery and has been altered to view star-forming regions [5,13]. 3D Slicer supports the FITS file format which is one of the most widely used formats [5]. There is very little documentation on 3D slicer as it is still in development. Its primary mode of visualization is 3D and it supports a variety of features such as, local data processing and it allows for rotation of the data cubes to encourage interactivity with the data [5]. 3D slicer is also an open source application, contributing to the extendibility of the application [5]. Although 3D Slicer is still in development, it allows for fast and flexible volume rendering and can be easily extended [5]. Due to its limited documentation there are no weaknesses mentioned for this tool except for that it is still in development.

2.6 FITS3D

FITS3D is a recently developed visualization tool created by Pragma Mohan, Christopher Hawkins, Roman Kaplauhk and Melanie Johnston-Hollit to allow for fast and interactive exploration of multiple 3D and 4D FITS files at the same time [3]. FITS3D supports the FITS file format and is still in beta testing [3] and therefore does not have a lot of documentation on it. FITS3D primary visualization modes are 3D and 4D. The features currently available in FITS3D are, local data processing, the exploration of multiple 3D and 4D FITS files at the same time, loading of multiple cubes simultaneously, real time histogram-based filtering, smoothing and rotations, and it allows for rapid changes to quality settings (this is also a strength of FITS3D) [3].

The strengths of FITS3D are that it supports fast and interactive exploration of multiple FITS files at the same time and it is faster than most visualization tools such as DS9 and Karma [3]. Due to FITS3D being in early stages there is no mention of its weaknesses and the only weakness it has is that it has not been tested completely because it is still in beta testing.

2.7 FITS2D

FITS2D is a visualization tool that was also created by Pragya Mohan, Christopher Hawkins, Roman Kaplaugh and Melanie Johnston-Hollit to visualize FITS files [3]. The application is still in beta testing and therefore there is not a lot of documentation on it. FITS2D is a 2D visualization tool with a primary feature of free form region extraction and annotation [3]. It also allows for local data processing, extracting of pixels as a new FITS file and the annotations can also be exported to a separate file [3]. Due to its poor documentation there is no information on the strengths and weaknesses of FITS2D and the only weakness we can think of is, like FITS3D, it has not been fully tested.

2.8 VisIVO

VisIVO was developed in collaboration between INAF (Catania Astrophysical Observatory) and CINECA in Italy. It was specifically designed for the Virtual Observatory and was developed for high-performance, multidimensional visualization of current and future very large-scale astrophysical data sets [4]. VisIVO supports multiple file formats namely, VOTables, FITS, HDF5, ASCII, raw binaries, CSV and native data format of popular gadget simulation code [14]. VisIVO has comprehensive documentation and is a 3D visualization tool.

VisIVO contains many different platforms such as VisIVO Desktop, VisIVO Server and VisIVO Web. The VisIVO Server contains a variety of tools such as VisIVO Importer, VisIVO Filter and VisIVO Viewer. Each platform and tool has its own set of features.

VisIVO Desktop is a stand-alone visualization application for standard PC's and VisIVO Server is a suite of software that is used to create custom 3D renderings of astronomical data from data tables. It is a new platform for large scale data sets [4].

VisIVO Importer converts data sets supplied by the user into VBTs (VisIVO Binary Tables). It also supports conversion from several different formats such as ASCII and CSV [4].

VisIVO Filter is a collection of data processing modules that can modify VBTs or create a new VBT from existing VBTs [4]. The operations of the main VisIVO modules consist of randomization, merge tables, extract sub regions, select rows, mathematical operations, point distribution, point property, coarse volumes, append tables, polar transformations, decimator, interpolate, show table, sigma contours, statistic and visual [4].

VisIVO Viewer is founded on the Visualization Toolkit version 5.2 [4]. VisIVO Viewer can be used to produce images in a sequence of azimuth, elevation and zooming that can be externally mounted to produce a movie. It also allows for the rendering of points, volumes and isosurfaces within a bounding box used to represent the coordinate system employed [4]. The standard output consists of four images which correspond to zoom factors and fixed camera positions, as well as another image that corresponds to user defined commands [4]. The different kinds of views supported by VisIVO Viewer include data point views, splotch views and volume views [4].

VisIVO Web is a World Wide Web portal that has been recently developed to provide VisIVO Server services to the scientific community [4].

VisIVO also has support for qualitative and quantitative data analysis as well as allows for animation not only of 2D scalar and vector fields, but 3D fields as well [23].

The strengths of VisIVO and its tools are that VisIVO Importer converts data without imposing any limit on size or dimensionality [4]. VisIVO server can be easily installed on any computing server and works with VisIVO Desktop [4]. VisIVO is fast at rendering files depending on how the user sets it up and it allows for scaling and the performance of VisIVO is dependent on the users RAM [4]. It also allows for complex tables to be loaded and manipulated as well as supports the merging of two table data structures to create a new one [14]. VisIVO has an interactive and modern interface and allows for active development [23].

The problems with VisIVO however, is that during the rendering process there is no response and before drawing there is no access to rendering options [23]. Another problem is that the use of the Look Up Table (LUT) editor is tiring and VisIVO has no crash recovery [23].

2.9 CyberSKA Viewer

CyberSKA viewer was developed for the Square Kilometer Array (SKA) and funded by Canada's Advanced Research and Innovation Network (CANARIE) as part of their Network Enabled Platform program [12]. CyberSKA was developed to help visualize the data being gathered by SKA and meet the requirements necessary to visualize the data being produced such as being able to handle large data volumes and data rates, complex and multi-purpose processing and analysis, as well as allowing it to be executed by a globally distributed team [12]. CyberSKA supports the FITS file format and is still under development, therefore there is not a lot of documentation on it. It is a 3D visualization tool and has a variety of features.

CyberSKAs features consist of interactive operations such as panning and zooming as well as histogram correction and colour map adjustments [12]. It also displays pixel value, region statistics and supports multiple coordinate systems, grids and a selection of frames for multi-dimensional images [12]. CyberSKA consists of a core functionality that then allows for plugins to be written using the API and added on top of the core to add additional functionality to it. This allows for custom tools to be developed to meet the astronomer's personal requirements.

The strengths of CyberSKA are it provides easy collaboration among astronomers all over the world, as well as providing access to various data, cloud computing resources and high performance computing [12]. CyberSKA does have a problem in that it has poor scalability.

2.10 Euro3D

Euro3D was developed by the Euro3D Training Network on Integral Field Spectroscopy under the funding of the European Commission [16]. It was originally designed towards meeting a few necessary requirements, namely, it must be able to display all the spectra stored on the file as a 2D image, display different spatial representations of the data from the stacked spectra representation, select spectra from the mapped data as well as allow the use of alternative representations of the data [18]. It is a visualization tool that uses the Euro3D data format, which was developed to try and create a unified data format, and is designed to handle large 3D data sets [6]. Euro3D has extensive documentation and is a 3D visualization tool.

Euro3D features are that it supports dead and/or low-efficiency spaxel removal from the data and has two routines namely, spexarith and spaxerith which allow for arithmetic operations between selected spectra and selected polychromatic maps [6]. Euro3D also performs spaxel selection and image reconstruction

[16]. It also allows for interaction with major astronomical data analysis packets such as, IRAF and PyRAF or IDL and implements a shared memory model in order to do so [18]. Euro3D also supports distributed data processing.

Euro3D's strength is that it is capable of handling complex geometries of mosaic pointings, without there being a need to add fake values, fill gaps or interpolate data to match a grid or pattern [16].

Euro3D does have the problem that it does not deallocate a fraction of the memory while reading in large data formats and therefore has a problem when it comes to processing huge datasets [16].

2.11 AstroVis

AstroVis was developed by the Department of Computer Science and Astrophysics, Cosmology and Gravity Centre (ACGC), Department of Astronomy at the University of Cape Town [2]. It was developed to allow for rapid visualization of large data files on standard desktop hardware [2]. AstroVis supports FITS file format and has very little documentation as it is fairly recent software. It is a 3D visualization tool and has features such as voxel slicing, panning and zooming [2]. AstroVis' architecture is that of a client server model however it currently works locally [2].

AstroVis' strengths are that it is efficient when viewing large scale data cubes, it allows the user to rapidly view the data and interact with the general observations as well as allowing for rapid qualitative analysis [2].

The problem with AstroVis however, is that in order to be able to perform quantitative analysis the user needs to wait for the entire data set to be read in [2].

3. DISCUSSION

Although all the tools mentioned above have a large variety of features there are still problems that these visualization tools face in general. One problem that all visualization tools face is the lack of scalability [17]. Another problem with visualization tools is that they all do one thing very well, however, there is no single software package that fully meets the requirements of all users [15]. The visualization tools also lack a standardised file format, for example the Euro3D visualization tool uses Euro3D data format but FITS3D uses FITS therefore in order to be able to use the two tools the data has to be converted from one file format to the other and in turn creating an unnecessary extra effort and extended processing time. The lack of uniformity of data formats becomes a problem when developing visualization tools [9]. Sánchez (2004) in E3D the Euro3D Visualization Tool also reinforces this by stating that a major problem when developing a visualization tool is the lack of a standardized format and this is the reason that Euro3D created their own format to try and standardise the data format. Most visualization tools also do not support remote data processing and with the size of the data currently, it is not realistic to store and process the data cubes locally, as most personal computers do not have the hard drive capacity to store these data cubes [13].

Table 1. Summary of all Tools Features, Strengths, Weaknesses, Data Format, Visualization Mode and Data Processing.

Visualization Tool (Ordered in terms of ranking)	Dimensions Displayed	Features	Strengths	Weaknesses	Local or Distributed Data Processing	Data Format
Karma	3D and 2D	<ul style="list-style-type: none"> • Functional concepts grouped into packages [1]. • Library is layered and each layer contains one or more packages [1]. • Packages at higher levels provide more power [1]. • Packages have no dependencies on other packages on any level [1]. 	<ul style="list-style-type: none"> • High usability [1]. • Obvious which system provides the greatest flexibility, power and ease of use [1]. • Mature 	<ul style="list-style-type: none"> • Slow [3]. • Cannot be used on a commodity laptop [3]. 	Distributed	FITS, Miriad n-dimensional array, PPM, Sun Rasterfile. [1]
GIPSY	3D	<ul style="list-style-type: none"> • Powerful analysis and galaxy modelling [11]. • Large variety of coordinate systems [11]. • Applications for modelling rotation curves, interactive inspection and characterization of 3D data [9]. 	<ul style="list-style-type: none"> • Mature [9]. • Powerful [9]. • Has good core functionality [9]. 	<ul style="list-style-type: none"> • Needs improvement on GUI [9]. • Needs better and more detailed documentation [9]. • Needs an easily installable and maintainable system [9]. 	Local	GIPSY. [19]

CASA	2D	<ul style="list-style-type: none"> • Functionality separated into two categories: tasks and tools [7]. • Data flagging [8]. • Playing animations by stepping through the z-axis [8]. • Overlaying of images [8]. • Supports non-interactive and interactive clean modes during imaging [8]. • Create multiple interactive plots at once [8]. • It is able to display multiple graphical user interfaces [8]. 	<ul style="list-style-type: none"> • Powerful [8]. • Tools provide flexibility and functionality [8]. • Tasks are user friendly [8]. • Allows for multiple interactive plots and GUI's [8]. 	<ul style="list-style-type: none"> • Does not support the entire region shapes and parameters [21]. • Viewer may not properly open saved region files [21]. 	Local	CASA, FITS. [8]
DS9	3D	<ul style="list-style-type: none"> • Supports FTP and HTTP access allowing for easy communication with other tools [10]. • Remote analysis [10]. • Has support for the Virtual Observatory [10]. • Built in help facility [10]. • Allows web based archive servers [10]. • Able to bin fits event files into a 3D data cube [10]. • Supports FITS multiple WCS [11]. • Toggle display of various coordinate systems [11]. • Support for composite regions and templates [11]. • Overlay display [11]. • Full support for loading, displaying, filtering and saving catalogues [11]. • Region annotation with text and simple shapes [3]. 	<ul style="list-style-type: none"> • Simple GUI [22]. • Simple installation process [22]. 	<ul style="list-style-type: none"> • Slow when processing large amounts of data [3]. • Cannot be used on a commodity laptop [3]. 	Distributed	FITS. [10]
3D Slicer	3D	<ul style="list-style-type: none"> • Open source to allow for extendibility [5]. • Used to view star forming regions [13]. • Allows for rotation of data cubes [5]. 	<ul style="list-style-type: none"> • Fast and flexible volume rendering [5]. • Can be easily extended [5]. 	<ul style="list-style-type: none"> • Still under development. 	Local	FITS. [5]
FITS3D	3D	<ul style="list-style-type: none"> • Exploration of multiple 3D and 4D FITS files at the same time [3]. • Loading of multiple cubes simultaneously [3]. • Real time histogram-based filtering, smoothing and rotations [3]. • Rapid changes to quality settings [3]. 	<ul style="list-style-type: none"> • Fast and interactive exploration of multiple 3D and 4D FITS files at the same time [3]. • Faster than most visualization tools [3]. • Rapid changes to quality settings [3]. 	<ul style="list-style-type: none"> • Still in Beta testing [3]. 	Local	FITS. [3]
FITS2D	2D	<ul style="list-style-type: none"> • Free form region extraction and annotation [3]. • Extracting of pixels as a new FITS files [3]. 		<ul style="list-style-type: none"> • Still in Beta testing [3]. 	Local	FITS. [3]

		<ul style="list-style-type: none"> Annotations can also be exported to a separate file [3]. 				
VisIVO	3D	<ul style="list-style-type: none"> Supports the merging of two table data structures to create a new one [14]. Supports loading and manipulating of complex tables [14]. Consists of multiple platforms namely VisIVO Desktop, VisIVO Server and VisIVO Web [4]. VisIVO Server contains a variety of tools such as VisIVO Importer, VisIVO Filter and VisIVO Viewer [4]. Support for qualitative and quantitative data analysis [23]. Allows for animation of both 2D and 3D scalar and vector fields [23]. 	<ul style="list-style-type: none"> Fast at rendering files [4]. Allows for scaling [4]. VisIVO Importer converts data sets supplied by the user into VBTs without imposing any limit on size or dimensionality [4]. VisIVO Server is easily installed on any computing server [4]. Performance of VisIVO is dependent on the users RAM [4]. Interactive and modern interface [23]. Allow for active development [23]. 	<ul style="list-style-type: none"> During rendering process there is no response [23]. Before drawing there is no access to rendering option [23]. Use of Look Up Table (LUT) editor is tiring [23]. No crash recovery [23]. 	Distributed	VOTables, FITS, HDF5, ASCII, raw binaries, CSV, native data format of popular gadget simulation code. [14]
CyberSKA	3D	<ul style="list-style-type: none"> Remote analysis. Panning and zooming [12]. Histogram correction [12]. Colour map adjustments [12]. Displays pixel value and region statistics [12]. Supports multiple coordinate systems, grids and selection of frame for multi-dimensional images [12]. Consists of a core functionality that then allows for plugins to be written using the API and added on top of the core to add additional functionality to it. 	<ul style="list-style-type: none"> Allows for collaboration among astronomers [12]. 	<ul style="list-style-type: none"> Poor scalability. 	Distributed	FITS. [12]
Euro3D	3D	<ul style="list-style-type: none"> Dead and/or low-efficiency spaxel removal from the data [6]. Spaxel selection [16]. Image reconstruction [16]. Allows for interaction with major astronomical data analysis packets such as IRAF and PyRAF or IDL and implements a shared memory model in order to do so [18]. Displays all the spectra stored on the file as a 2D 	<ul style="list-style-type: none"> Capable of handling complex geometries of mosaic pointings, without there being a need to add fake values, fill gaps or interpolate data to match a grid or pattern [16]. 	<ul style="list-style-type: none"> Does not deallocate a fraction of the memory while reading in large data formats [16]. Problem when processing huge datasets [16]. 	Distributed	Euro3D. [6]

		<ul style="list-style-type: none"> image [18]. Displays different spatial representations of the data from the stacked spectra representation [18]. Selects spectra from the mapped data [18]. Allows the use of alternative representations of the data [18]. 				
AstroVis	3D	<ul style="list-style-type: none"> Voxel slicing [2]. Panning and zooming [2]. Architecture of a client side model however it currently works locally [2]. 	<ul style="list-style-type: none"> Allows the user to rapidly view the data and interact with the general observations [2]. Efficient at the viewing of large scale data cubes [2]. Rapid qualitative analysis [2]. 	<ul style="list-style-type: none"> To perform quantitative analysis the user needs to wait for the entire data set to be read in [2]. 	Local	FITS. [2]

Table 1 above gives a summary of all the tools features, strengths, weaknesses, data format, visualization mode and data processing type, so that these tools can be easily compared to one another. For the next generation of visualization tools, we can use the above information in the table to determine what aspects to keep and try and improve on. Features such as overlaying of images is an important feature to try implement as it is not implemented by many of the tools and will allow for easy comparison between data sets. This also gives the option of allowing for a comparison between optical and radio data. Another feature to try and incorporate into future visualization tools is annotation and region selection as it will allow astronomers to mark interesting regions for further study. More features such as panning and zooming to improve interactivity, distributed data processing to be able to process large data cubes without having to take up the space on your hard drive, will also have to be implemented. In terms of file support FITS and HDF5 are the most common file formats currently and therefore the application will need to support these.

When designing the next generation of visualization tools not only do you need to think of features but also the challenges of developing visualization tools for astronomical data. Hassan and Fluke (2011) mention that currently in the Petascale Astronomy Era there are six main challenges of visualization, namely, support for quantitative visualization, effective handling of large data sets, discovery in low signal-to-noise data, better human computer interaction, better workflow intergration and encouragement for adoption of 3D scientific visualization techniques. All these challenges need to be taken into account when developing the future visualization tools for astronomical data so that a new tools with the same problems are not developed.

4. CONCLUSIONS

The main discovery of the analysis of these tools is the fact that there is a large number of powerful tools for visualizing astronomical data cubes, however, there are no tools available that can meet all the requirements for all the users in the astronomical community.

Visualization tools in astronomy seem to have a long way to go in order to develop a tool that can be easily used by everyone, meet all their needs, support a common unified file format such as FITS or HDF5, support remote data analysis, perform all the functions needed to obtain the necessary information about the provided data and is able to process the vast amounts of diverse data being produced by the telescopes of today. The astronomical and computer science community needs to work together to develop such a tool so that more effective and efficient processes can be performed on the data so that the new data being produced can be analyzed more carefully and more information can be obtained from it.

5. REFERENCES

- [1] Gooch, R. 1996. KARMA: a visualization test-bed. **Proceedings of *Astronomical Data Analysis Software and Systems (ADASS) V, ASP Conference Series, Vol. 101, G. H. Jacoby, and J. Barnes, eds.* 80-83.
- [2] Perkins, S. et al. 2014. Scalable desktop visualisation of very large radio astronomy data cubes. *New Astronomy.* 30, 1–7.
- [3] Mohan, P. et al. 2016. Three Tools to Aid Visualisation of FITS Files for Astronomy. 2003–2006.
- [4] Becciani, U. et al. 2010. VisIVO—Integrated Tools and Services for Large-Scale Astrophysical Visualization. *Publications of the Astronomical Society of the Pacific.* 122, 887, 119–130.
- [5] Borkin, M. et al. 2007. Application of medical imaging software to 3D visualization of astronomical data. **Proceedings of *Astronomical Data Analysis Software and Systems (ADASS) XVI, ASP Conference Series, Vol. 376, R. A. Shaw, F. Hill, and D. J. Bell, eds.* 612-624.
- [6] Sánchez, S.F. et al. 2004. E3D, the Euro3D visualization tool II: Mosaics, VIMOS data and large IFUs of the future. *Astronomische Nachrichten.* 325, 2, 171–174.
- [7] McMullin, J.P. et al. 2007. CASA architecture and applications. **Proceedings of *Astronomical Data Analysis*

- Software and Systems (ADASS) XVI, ASP Conference Series, Vol. 376, R. A. Shaw, F. Hill, and D. J. Bell, eds. 127-130.*
- [8] Jaeger, S. 2008. The Common Astronomy Software Application (CASA). **Proceedings of * Astronomical Data Analysis Software and Systems (ADASS) XVII, ASP Conference Series, Vol. 394, R. W. Argyle, P. S. Bunclark, and J. R. Lewis, eds. 623-626.*
- [9] Ruiz, J.E. et al. 2009. GIPSY 3D: Analysis, Visualization and VO Tools for Datacubes. **Proceedings of * Astronomical Data Analysis Software and Systems (ADASS) XVIII, ASP Conference Series, Vol. 411, D. Bohlender, D. Durand, and P. Dowler, eds., 406-409.*
- [10] Joye, W.A. and Mandel, E. 2003. New features of **Proceedings of * Astronomical Data Analysis Software and Systems (ADASS) XII, ASP Conference Series, Vol. 295, H. E. Payne, R. I. Jedrzejewski, and R. N. Hook, eds.489-492.*
- [11] Joye, W.A. 2006. New features of SAOImage DS9. **Proceedings of * Astronomical Data Analysis Software and Systems (ADASS) XV, ASP Conference Series, Vol. 351, C. Gabriel, C. Arviset, D. Ponz, and E. Solano, eds. 574-576.*
- [12] Kiddle, C. et al. 2011. Looking Towards the Future of Radio Astronomy with the CyberSKA Collaborative Portal. **Proceedings of * Astronomical Data Analysis Software and Systems (ADASS) XX, ASP Conference Series, Vol. 442, I. N. Evans, A. Accomazzi, D. J. Mink, and A. H. Rots, eds. 669-672.*
- [13] Goodman, A.A. 2012. Principles of high-dimensional data visualization in astronomy. *Astronomische Nachrichten. 333, 5-6, 505-514.*
- [14] Comparato, M. et al. 2007. Visualization, Exploration, and Data Analysis of Complex Astrophysical Data. *The Publications of the Astronomical Society of the Pacific. 119, 898-913.*
- [15] Hassan, A. and Fluke, C.J. 2011. Scientific visualization in astronomy: Towards the petascale astronomy era. *Publications of the Astronomical Society of Australia. 28, 2, 150-170.*
- [16] Sánchez, S.F. 2004. E3D, the Euro3D visualization tool I: Description of the program and its capabilities. *Astronomische Nachrichten. 325, 2 (2004), 167-170.*
- [17] Kent, B.R. 2013. Visualizing Astronomical Data with Blender. *Publications of the Astronomical Society of the Pacific. 125, 928, 731-748.*
- [18] Sánchez, S.F. 2004. E3D, The Euro3D Visualization Tool. **Proceedings of * Astronomical Data Analysis Software and Systems (ADASS) XIII, ASP Conference Series, Vol. 314, F. Ochsenbein, M. Allen, and D. Egret, eds. 517-521.*
- [19] 1 Introduction: 2016. <http://www.atnf.csiro.au/computing/software/karma/user-manual/node2.html#pubpartialdata>. Accessed: 2016- 04- 20.
- [20] Linking Visualization and Understanding in Astronomy #AAS223 #AASviz: 2016. <http://www.astrobetter.com/blog/2014/01/05/linking-visualization-and-understanding-in-astronomy-aas223/>. Accessed: 2016- 04- 20.
- [21] CASA: 2016. https://casa.nrao.edu/release_ki.shtml. Accessed: 2016- 04- 25.
- [22] Joye, W.A. and Mandel, E. 2005. The Development of SAOImage DS9: Lessons Learned from a Small but Successful Software Project. **Proceedings of * Astronomical Data Analysis Software and Systems (ADASS) XIV, ASP Conference Series, Vol. 347, P. L. Shopbell, M. C. Britton, and R. Eber., eds. 110-113.*
- [23] 3D Visualization: VisIVO: 2016. <http://wwwmpa.mpa-garching.mpg.de/~mselig/visivo/visivo.html>. Accessed: 2016- 04- 25.
- [24] Blyth, S. 2016. Would you be able to easily rank the following visualization tools in order of most widely used in astronomy?.