3D visualization and analysis of HI in and around galaxies

Punzo, Davide

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version
Publisher's PDF, also known as Version of record

Publication date:
2017

Link to publication in University of Groningen/UMCG research database

Citation for published version (APA):

Copyright
Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

Take-down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): http://www.rug.nl/research/portal. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.

Download date: 31-01-2020
The role of 3D interactive visualization in blind surveys of H I in galaxies

— D. Punzo, J.M. van der Hulst, J.B.T.M. Roerdink, T.A. Oosterloo, M. Ramatsoku, M.A.W. Verheijen —

Chapter 2. The role of 3D interactive visualization in blind surveys of H I in galaxies

Abstract

Upcoming H I surveys will deliver large datasets, and automated processing using the full 3D information (two positional dimensions and one spectral dimension) to find and characterize H I objects is imperative. In this context, visualization is an essential tool for enabling qualitative and quantitative human control on an automated source finding and analysis pipeline. We discuss how Visual Analytics, the combination of automated data processing and human reasoning, creativity and intuition, supported by interactive visualization, enables flexible and fast interaction with the 3D data, helping the astronomer to deal with the analysis of complex sources. 3D visualization, coupled to modeling, provides additional capabilities helping the discovery and analysis of subtle structures in the 3D domain. The requirements for a fully interactive visualization tool are: coupled 1-D/2D/3D visualization, quantitative and comparative capabilities, combined with supervised semi-automated analysis. Moreover, the source code must have the following characteristics for enabling collaborative work: open, modular, well documented, and well maintained. We review four state-of-the-art, 3D visualization packages assessing their capabilities and feasibility for use in the case of 3D astronomical data.
2.1 Introduction

The Square Kilometre Array (SKA) and its precursors are opening up new opportunities for radio astronomy in terms of data collection and sensitivity. Two types of blind surveys are planned with SKA-pathfinders:

1. shallow (very large sky coverage): WALLABY with ASKAP (Johnston et al., 2008; Duffy et al., 2012), shallow and medium-deep Apertif surveys with the WSRT (Verheijen et al., 2009).

2. deep (high sensitivity, small solid angle): CHILES with the J-VLA (Perley et al., 2011; Fernandez et al., 2013); LADUMA with MeerKAT (Jonas, 2009; Holwerda et al., 2012) and DINGO with ASKAP (Johnston et al., 2008; Duffy et al., 2012).

The first type of HI survey will detect $\sim 10^3$ sources weekly, of which 0.2% will consist of well resolved sources, 6.5% will have a limited number of resolution elements, and 93% will at best be marginally resolved (Duffy et al., 2012). This predicted weekly data rate is high, and fully automated pipelines will be required for processing the data (see section 2.3). The first and second category of sources will contain a wealth of morphological and kinematic information. However, in cases with complex kinematics it will be difficult to extract all information in a controlled and quantitative way (Sancisi et al., 2008; Boomsma et al., 2008). Therefore, manual analysis of a subset of the resolved sources will still be required. In fact, manual processing will be very useful for obtaining a deeper insight in particular features of the data (e.g., tails and extra-planar-gas; see section 2.3.3). It will also enhance possible improvements to the automated pipelines. For example, it can play a major role in the development and training of machine learning algorithms (e.g., deep learning: Ball and Brunner, 2010; Graff et al., 2014) for the automated analysis data, in particular in the era of the full SKA data (see section 2.2.6).

The SKA pathfinders will provide a wealth of data, but the expected exponential growth of the data has created several data challenges. We will present a preview of the data that Apertif will deliver to the community in the near future and discuss the importance of visualization for the analysis of radio data in the upcoming surveys era. Our discussion will be based on existing mosaics acquired with the Westerbork Synthesis Radio Telescope (WSRT), which are representative of the daily image data rate provided by future blind HI surveys.
2.1.1 WSRT and the Apertif data

The WSRT consists of a linear array of 14 antennas with a diameter of 25 meters arranged on a 2.7 km East-West line located in the north of the Netherlands. The Apertif phased array feed system is an upgrade to the WSRT which will increase the field of view by a factor of 30 (Verheijen et al., 2008; Oosterloo et al., 2010), which allows a full inventory of the northern radio sky complemented by a wealth of optical, near-IR data, and other radio observatories such as the Low-Frequency Array (LOFAR). Part of the Apertif surveys will be a medium deep blind survey of a few hundred square degrees with a $3 \sigma$ column density depth of $2 - 5 \times 10^{19}$ cm$^{-2}$.

A full 12 hour integration will provide $\sim 2.4$ TB of complex visibilities sampling a $3^\circ \times 3^\circ$ region of the sky and the data reduction that follows will generate three dimensional datasets of the HI line emission, with axes right ascension (RA or $\alpha$), declination (DEC or $\delta$), and frequency ($\nu$) or recession velocity ($v$). The typical size of a data-cube will be $2048 \times 2048$ pixels for the spatial coordinates (each pixel covers 5 arcsec) and 16384 spectral channels, which correspond to 16384 pixels in the third dimension covering a bandwidth of 300 MHz ($\sim 60,000$ km/s). The disk storage needed for each data-cube is about 0.25 TB, assuming a single Stokes component, I, and 32 bits per pixel format. The final product after observing the northern sky will be of the order of 5 PB of data-cubes.

Examining these numbers it is clear that the storage, data reduction, visualization and analysis to obtain scientific results requires the development of new tools and algorithms which must exploit new solutions and ideas to deal with this large volume of data. The Tera-scale volume of these datasets produces, in fact, both technical issues (e.g., dimension of the data much larger than the available random access memory (RAM) on a normal workstation) and visualization challenges (i.e., the presence in each dataset of a large number of small sources with limited signal-to-noise ratio (SNR)).

2.1.2 Data visualization

Traditionally visualization in radio astronomy has been used for:

1. finding artifacts due to an imperfect reduction of the data;

2. finding sources and qualitatively inspect them;
3. performing quantitative and comparative analysis of the sources.

In this paper we will focus mainly on the connection between interactive visualization and the automated source finder and analysis pipeline (2); and the importance of interactive, quantitative and comparative visualization (3). We will not discuss visualization of artifacts (1) resulting from imperfections in the data. Artifacts can arise from several effects: Radio Frequency Interference (RFI), errors in the bandpass calibration, or errors in the continuum subtraction. Volume rendering can help localize such artifacts, but in that case visualization is envisaged to play the role of assisting quality control of the products of an automated calibration pipeline.

In section 2.2 we give an overview of the past and current visualization packages and algorithms, with a focus on radio astronomy. We highlight the 3D nature of the H1 data in section 2.3. The definition of the requirements for a fully interactive visualization tool is given in section 2.4. Finally, in section 2.5 we review state-of-the-art visualization packages with 3D capabilities. Our aim is to define the basis for the development of a 3D interactive visualization tool.

2.2 Scientific visualization

Scientific visualization is the process of turning numerical scientific data into a visual representation that can be inspected by eye. The concept of scientific visualization born in the 1980’s (McCormick et al., 1987; Frenkel, 1988; DeFanti et al., 1989). Its role was not relegated to only presentation (Roerdink, 2013). The interactive processing of the data, the imaging and analysis, including qualitative, quantitative and comparative stages, is crucial for achieving a deep and complete knowledge.

In this section we provide background information about past visualization developments in astronomy, scientific visualization theory, visualization hardware and the software terminology used in this paper.

2.2.1 Visualization in astronomy

One of the first systematic radio astronomy visualization trials was undertaken by Ekers and Allen (1975) (see also Allen, 1979; Sedmak et al., 1979; Allen, 1985). They investigated techniques for displaying single-image
datasets, including contour display, ruled surface display, grey scale display, and pseudo-color display. They also discussed techniques for the display of multiple image datasets, including false-color display and cinematographic display.

At the beginning of the 1990’s, Mickus et al. (1990b), Domik (1992), Mickus et al. (1990a), Domik and Mickus-Miceli (1992), and Brugel et al. (1993) developed a visualization tool named the Scientific Toolkit for Astrophysical Research (STAR). STAR was a prototype resulting from the development of an user interface and the implementation of visualization techniques suited to the needs of astronomers at that time. These included display of one- and two-dimensional datasets, perspective projection, pseudo-coloring, interactive color coding techniques, volumetric data displays, and data slicing.

Recently, both Hassan and Fluke (2011) and Koribalski (2012) pointed out the lack of a tool that can deal with large astronomical data-cubes. In fact, the current astronomy software packages are characterized by a window interface for 2D visualization of slices through the 3D data-cube; in some cases limited 3D rendering is also present. Moreover, they can exploit only the resources of a personal computer which imposes strong limitations on the available RAM and processing power. Stand-alone visualization tool examples are KARMA (Gooch, 1996), SAOImage DS9 (Joye, 2006), VisIVO (Comparato et al., 2007; Becciani et al., 2010). Other viewers exist but are embedded in reduction and analysis packages: GIPSY (van der Hulst et al., 1992; Vogelaar and Terlouw, 2001), CASA (McMullin et al., 2007) and AIPS (Greisen, 2003). In addition, S2PLOT (Barnes and Fluke, 2008) is a programming library that supports creation of customized software for interactive 3D visualizations. A recent development is the use of the open source software Blender for visualization of astronomical data (Kent, 2013; Taylor et al., 2014), but this application is more suitable for data presentation rather than interactive data analysis.

From the inventory of the current state of-the-art we conclude that the expected exponential growth of radio astronomy data both in resolution and field of view has created a necessity for new visualization tools. In the meantime much development has taken place in computer science and medical visualization. We review relevant software from these areas in sections 2.3 and 2.5.
2.2.2 3D visualization

First investigations of the suitability of 3D visualization for radio-astronomical viewers date back to the beginning of the 90’s (Norris [1994]). Already at that time it was clear that a 3D approach can provide a better understanding of the 3D domain of the radio data. The type of data slicing commonly used (i.e., channel movies), forces the researcher to remember what was seen in other channels and requires a mental reconstruction of the data structure. The major advantage of a 3D technique is an easier visual identification of structure, including faint features extending over multiple channels. A crucial point made by Norris is that presenting the results qualitatively is fine for data inspection, but that interactive and quantitative hypothesis testing requires quantitative visualization.

In the last twenty years hardly any new 3D visualization tools were developed for examining 3D radio astronomical data. In the middle of the 1990’s, Oosterloo (1995) investigated porting direct volume rendering techniques to radio astronomy visualization. He analyzed the features and the issues related to a ray casting algorithm (a massively parallel image-order method; see Roth [1982]), pointing out, in general, the advantages and drawbacks of the 3D visualization. He could, however, not develop a run-time 3D interactive software package due to the lack of available computational resources.

2.2.3 Volume rendering

3D visualization is the process of creating a 2D projection on the screen of the 3D objects under study. This process is called volume rendering. The rendering methods are divided in two principal families: indirect volume rendering (or surface rendering) and direct volume rendering. The first approach fits geometric primitives through the data and then it renders the image. It requires a pre-processing step on the dataset, then after the pre-processing a quick rendering is possible. Fitting geometric primitives, however, may introduce noise errors due to rendering artifacts. Moreover, not all datasets can be easily approximated with geometric primitives and the HI sources fall into this class because they do not have well defined boundaries. Furthermore, in a HI data-cube the signal-to-noise is usually low. For example, the galaxies in the WHISP survey (van der Hulst et al., 2001), have average signal-to-noise of $\sim 10$ in the inner parts and $\sim 1$ in the outer parts. This makes indirect volume rendering inefficient. Direct
volume rendering methods directly render the data defined on a 3D grid (each element of the grid is called a \textit{voxel}), therefore it requires more computations to process an image. Several direct rendering solutions exist and they are classified as: 1) Object order methods, requiring an iteration over the voxels which are projected on the image plane; 2) Image order methods, which instead iterate over the pixels of the final rendered image and have the algorithm calculate how each voxel influences the color of a single pixel. 3) Hybrid methods, a combination of the first two. It must be noted that during the process of direct volume rendering the depth information can be mixed depending on the projection method used (i.e., \textit{maximum}, \textit{minimum}, and \textit{accumulate}). By rotating or the use of 3D hardware the human user is able to mentally connect the various frames and to register the proper 3D structures. For a detailed review of the state of-the-art and for more information we refer to the Visualization Handbook \cite{HansenBook2004} and the VTK book \cite{VTKBook2006}.

\subsection*{2.2.4 Out-of-core and in-core solutions}

The rendering software can exploit an out-of-core or an in-core solution. Out-of-core solutions are optimized algorithms designed to handle datasets larger than the main system memory by utilizing secondary, but much slower, storage devices (e.g., hard disk) as an auxiliary memory layer. These algorithms are optimized to efficiently fetch or pre-fetch data from such secondary storage devices to achieve real-time performance. They usually utilize a multi-resolution data representation to facilitate such a fast fetch mechanism and to support different available output resolutions based on the limitations in terms of the processing time and the available computational resources \cite{Rusinkiewicz2000, Crassin2009}.

The in-core solution can achieve very fast memory transfer because it does not need to access the data stored on a hard disk continuously. In fact, it assumes that the data are stored in the main system memory, ready for processing. Of course, in this case the main limitation is the size of the available RAM.
2.2.5 3D hardware

The use of 2D input and output hardware limits the possible interaction with a 3D representation. 3D input devices (such as 3D mouse or pointer) can naturally solve this problem. Moreover, coupling them to a 3D output device such as a 3D monitor, a CAVE virtual environment, etc., can remove the difficulty of positioning a 3D cursor in a 3D space. In fact, in this case, the user can see the real 3D movement, instead of the projection on a 2D screen. However, virtual reality has never been widely used in the researchers’ daily work due to the dependence on very expensive hardware not available on the common computer market.

Recently, two new very promising devices, the Leap Motion (an input device that tracks the hands in 3D)\(^1\) and the Oculus Rift (a 3D output device with a full immersion virtual reality, VR, experience)\(^2\), appeared that can change this situation, because they are aimed for the gaming market, and therefore will be rather cheap.

This hardware could enhance new interaction perspectives with volumetric data using a desktop solution. Recent efforts have been undergoing to bring these powerful VR capabilities by Ferrand et al. (2016); Fluke and Barnes (2016). Although their results are promising, we will exclude such hardware solutions from our visualization discussion because the success and therefore the maintainability of a visualization solution based on them, which depends on the gaming market, is still uncertain. Moreover, from the point of view of interface design, the use of this new hardware creates the need to develop new interface concepts. The equivalent expertise that exists for classical interfaces such as mouse and keyboard is, however, still missing. This does not exclude that in the coming years virtual reality may become very popular and stimulate many developers to experiment with the Leap Motion and the Oculus Rift (or HTC Vive)\(^3\) or future 3D hardware.

2.2.6 Visual Analytics

In the SKA era manual inspection and analysis of even only a subset of data will be extremely hard to achieve. Machine learning (e.g., deep learning: Ball and Brunner, 2010; Graff et al., 2014) will be needed for classification

---

1. \url{https://www.leapmotion.com/}
2. \url{http://www.oculus.com/}
3. \url{https://www.vive.com/}
Chapter 2. The role of 3D interactive visualization in blind surveys of H I in galaxies

of the different components of a galaxy (de la Calleja and Fuentes, 2004; Banerji et al., 2010). However, the reliability of the analysis by machine learning heavily depends on the input for the training session (Kuminski et al., 2014). Discovering interesting relations, structures, and patterns in very large and high-dimensional data spaces needs the combination of automated data processing with human reasoning, creativity and intuition, supported by interactive visualization. Human assessment remains essential for understanding the behavior of automatic algorithms and for visual quality control. As the available data grow, effective and efficient techniques are essential to increase our insight in the underlying structures and processes.

Combining interactive visualization with analytic techniques (machine learning, statistics, data mining) has grown into a field of its own: Visual Analytics (Thomas and Cook, 2005; Keim et al., 2010). It aims to fully integrate human expertise in the human-machine dialogue to steer the sense-making process. Visual analytics supports collaborative exploration and decision making by combining fast access to large distributed databases, data integration, powerful computing infrastructures, and interactive visualization facilities (e.g., large touch displays). Astronomy is an exciting and extremely demanding testfield for new visual analytics techniques. Data availability, storage and distribution are well covered. Expert knowledge is available to validate algorithmic approaches. Data-set dimensionality (dimension $d = 10 \ldots 100$) and sizes ($> 10^9$ elements) make scalability extremely difficult to achieve. Extracting meaningful relations across the entire set of data dimensions is inherently hard for data of high dimensionality (Bertini, 2011). Integrating data sources, data-reduction algorithms, and expert knowledge to effectively and efficiently answer domain-specific questions is an open challenge. Visual analytics advocates a mixed approach: automatically search datasets for potentially meaningful patterns, and interactively steer data reduction and visualization.

### 2.3 Visualization of H I datasets

The domain of future radio surveys, such as those planned with Apertif, will fall in the Big Data domain for two reasons:

1. a data-cube will have dimensions of $2048 \times 2048 \times 16384 \sim 68.7 \times 10^9$ (0.25 TB). The data rate is $\sim 10$ cubes/week;
2. each data-cube will contain $\sim 100$ sources, i.e. galaxies, of relatively small typical size ($\sim 10^5$ voxels) in the observed data volume of $\sim 10^{11}$ voxels.

A very important step is to condense this vast amount of data collected by the surveys into a much smaller catalog of interesting regions, the sources, and their properties. This is done by examining the data itself. If done manually, the astronomers have to explore the whole data set using visualization software in order to identify the sources.

2.3.1 Visualization and source finding

For illustrative purposes, we consider a mosaicked data-cube that serves as a pilot training set for future, single Apertif pointings (Ramatsoku et al., 2016). The mosaic is built from 35 individual WSRT pointings in a hexagonal grid, directed towards a region in the sky where a filament of the Perseus-Pisces Supercluster (PPScl) crosses the plane of the Milky Way. The data-cube covers a sky area of $2.4^\circ \times 2.4^\circ$ centered at $\alpha = 72^\circ$ and $\delta = 45^\circ$. The redshift range is $cz = 2000 - 17000$ km/s. The resulting data-cube has dimension $2300 \times 2300$ pixels for the spatial coordinates and 1717 pixels in the velocity dimension. This is $\sim 10$ times smaller in the velocity (frequency) dimension than a single Apertif pointing, but the spatial resolution, velocity resolution and sensitivity are comparable. The number of objects is also comparable as Perseus-Pisces is an over-dense region. The $\sim 200$ sources comprise $\lesssim 1\%$ of the data volume. The minimum column density detected is $\sim 6.4 \times 10^{19}$ cm$^{-2}$ at the $3\sigma$ level over a velocity range of 16.5 km/s.

The three-dimensional visual representation of the mosaic in Fig. 2.1 immediately highlights the sources’ shape and position in the data-cube. Moreover, interactivity such as rotation, zooming and panning, and editable color transfer functions greatly support manual identification of the sources in the data-cube.

An interactive in-core ray casting algorithm running on a cluster of Graphic Processing Units (GPUs) has been proposed by Hassan et al. (2013) for the visualization of Tera-scale radio astronomy data-cubes. In general, many large visualization software tools are in development in the context of computer science and medical imaging. Some notable examples follow:
Figure 2.1 – Two representations of the H\textsc{i} in galaxies in a filament of the Perseus-Pisces Supercluster (PPScI) are shown. In the top-left panel, the rendering of the full data-cube with a maximum intensity projection method is illustrated. In the top-right panel, the data-cube after a semi-automated procedure, performing, with GIPSY routines, the smooth and clip procedure as implemented in Serra et al. (2015), is shown. In both cases it is possible to see a large number of sources, but many of them are hardly visible in the top-left panel. In the bottom panels, two zooms are provided. Smoothing has been applied at the bottom-left sub-cube revealing some of the sources (circled) easily visible in the bottom-right panel.
2.3. Visualization of HI datasets

1. in-core solutions exploiting parallel computing on a cluster: Morelanda et al. (2007) (i.e., ParaView) and Vo et al. (2011);

2. out-of-core solutions: Crassin et al. (2009) (i.e., GigaVoxels) and Hadwiger et al. (2012).

In the case of visualization of HI in galaxies, it is, however, unlikely that visualization of the full data-cube will be used for finding sources for the following reasons:

1. the size of the HI blind survey data volume and the number of sources, as illustrated in Fig. 2.1, prohibit a manual approach even when using very powerful interactive visualization tools;

2. radio data are intrinsically noisy, and most sources are faint and often extended. Spatial and/or spectral smoothing increase the signal-to-noise ratio depending on the source structure. In fact, smoothing is applied on multiple spatial and spectral scales to ensure that sources of different size are extracted at their maximum integrated signal-to-noise ratio. In Fig. 2.1 two visual representations of the PPScl data, respectively before and after the source finder step, are shown. In both cases it is possible to see a large number of sources, but many of them are hardly visible in the original data-cube because they are drowned in the noise. Manual operations such as zooming, changing the color function, and smoothing help the observer in identifying the sources visually. This will, however, take a large amount of time if done manually and repeatedly on the full Tera-scale data-cubes which will be delivered at a rate of 1-2 per day. A suitable framework can be achieved but it will require both major investments in hardware resources and dedicated radio astronomers performing the manual inspection.

3. interactive rendering of $\sim 10^{11}$ voxels using an in-core solution, such as Hassan et al. (2013) have demonstrated, requires considerable resources for hardware and maintenance, not affordable by typical research groups or major observatories. An out-of-core solution can reduce the financial demands on hardware. However, the development itself of such a solution requires a huge programming effort due to

\[\text{https://github.com/Amr-H-Hassan/GraphTIVA}\]
many challenges related to the I/O bandwidth limits. We refer to [Crassin et al., 2009] and [Hadwiger et al., 2012] for a detailed description of state-of-the-art out-of-core visualization algorithms, including CPU-GPU memory transfer solutions. Note, however, that the last two rendering pipelines cited are not publicly available yet.

Automated pipelines have been developed to extract the source information from the data collected [Whiting, 2012; Serra et al., 2015]. Their goal is to find all reliably detectable extragalactic H I objects in the observed data volume, and to determine the properties of these objects, that is:

1. the galaxies, i.e., the regularly rotating gas disks;
2. additional H I structures such as extra-planar-gas and tails. These are crucial for understanding the detailed balance between gas accretion and gas depletion processes, as well as their dependence on the environment, and for obtaining the full picture of galaxy evolution. For example, extra-planar-gas data can be used to quantitatively constrain the gas accretion and depletion processes (see section 2.3.3). Another example is the presence of tails in the data. Tails can be produced by tidal interactions between galaxies (Fig. 2.3) or by ram pressure stripping [Oosterloo and van Gorkom, 2005], and are strong indications for these processes. Deciding which process is important requires detailed inspection and modeling of the features discovered in the data. We refer to [Sancisi et al., 2008] for a full review of the state-of-the-art of HI observations and their interpretation. These features are located in the vicinity of the galaxies and have low column densities and low signal-to-noise;
3. the faint H I in the cosmic web such as HI filaments between galaxies. This emission is expected to have very low column density and very low signal-to-noise in a single resolution element, so will be difficult to detect. It is probably extended, following the large-scale structure, so the signal-to-noise could be increased by smoothing. This is, however, unlikely to be sufficient for detection (see below).

For inspecting (1) and (2), visualization techniques can be used in the following approach: high-dimensional visualization (e.g., 3D scatter plots) of the parameters provided by the pipeline and stored in catalogs (such as
position, flux, flux error, degree of asymmetry, velocity width, integrated profile asymmetry, etc.) gives an overview of the data and their 3D domain (see section 2.4.4). Then, manual inspection will be performed for only a subset of sources, which can be delivered to a visualization analysis package with full rendering capabilities for further analysis (see section 2.3.3). In the case of (3) we should point out that future observations with the SKA precursors, such as Apertif and ASKAP, will not achieve the sensitivity to detect the cosmic web. The neutral fraction of cosmic web filaments is expected to be very low, leading to HI column densities $\lesssim 10^{18}$ cm$^{-2}$, Braun and Thilker (2004); Ribaudo et al. (2011). We therefore do not focus on such low level and extended emission.

2.3.2 Automated pipelines and human intervention.

Automated pipelines will be responsible for finding the sources, measuring parameters that give an indication of the properties of a source and creating catalogs. Source finders are designed to automatically detect all the sources in the field. In order to do that, source finders must employ an efficient mechanism to discriminate between such interesting regions and the noise. The peak flux, total flux, and number of voxels are parameters that can be used to determine the completeness and reliability of detected sources when examining both positive and negative detections (Serra et al., 2012a). Due to the complex 3D nature of the sources (Sancisi et al., 2008) and the noisy character of the data, it is, however, not trivial to construct a fully automated and reliable pipeline. A review of the current state-of-the-art is given by Popping et al. (2012), who describe the issues connected with the noisy nature of the data, and the various methods and their efficiency. In addition, automated source characterization and measurement of source parameters are required for producing catalogs with science-ready products. Human inspection will be necessary for quality control of the results from the pipelines and in particular for investigating complex cases. The human mind, in fact, is a very powerful diagnostic instrument which can naturally recognize (source) structures in the data. For example, in a significant number of cases, it will be very difficult to automatically retrieve information about particular features such as tidal tails or stripped HI. Apertif most likely will deliver 2 or 3 of these cases every day (estimate based on the data shown in Fig. 2.1). The analysis of these will still be done manually and visualization will still play a major role. In fact,
automated algorithms are built on the knowledge acquired during the manual approach (see section 2.2.6 for the role of interactive visualization and machine learning in visual analytics). Moreover, coupling visualization tools with semi-automated data analysis techniques is necessary in order to improve the inspection itself.

The subcubes containing the sources detected by source finders will be relatively small with maximum sizes of $512 \times 512 \times 256 \sim 0.067 \times 10^9$ voxels, reducing the local storage, I/O bandwidth, and computational demand for visualization (easily achievable on a modern computer).

### 2.3.3 Visualization and source analysis

In this paragraph we will show in detail, using visualization examples, the character of the 21-cm radio emission of galaxies and the benefits and drawbacks of adopting 3D visualization, as pointed out already by Norris (1994) and Oosterloo (1995) (see also Goodman et al., 2009).

The use of 3D visualization of HI in galaxies is still in its infancy. Existing astronomical 3D visualization tools lack interactivity and the ability to perform quantitative analysis. The lack of interactivity is mainly a result of the lack of computing power to date, as volume rendering is computationally expensive. Moreover, the use of 2D input and output hardware limits the interaction with a 3D representation (see section 2.2.5). Therefore, the interpretation of a 3D visual representation has never been investigated thoroughly. An additional complication is that the 3D structure of the HI in a cube is not in a 3D spatial domain. The third axis represents velocity and thus the 3D rendering delivers a mix of morphological, kinematical and geometrical information. Therefore, 3D visual analytics has never been developed for HI data. These are the main reasons why the development of 3D visualization as a tool for inspecting, understanding and analysing radio-astronomical data has been slow. Currently available hardware, e.g. GPUs, now enable interactive volume rendering, stimulating further development.

3D visualization techniques can provide many insights about the source under study. In Fig. 2.2 the three-dimensional visualization of a particular source in the PPScl filament, discussed in section 2.3, shows a 3D view of its HI distribution and kinematics providing an immediate overview of the structures in the data. Two main components are visible in Fig. 2.2: a central body, which is the regularly rotating disk of the galaxy, and a
2.3. Visualization of H\textsubscript{I} datasets

Figure 2.2 – Three views of the volume rendering of a particular source in the PPScl filament are shown. The optical counterpart, WEIN069, has been observed by Weinberger et al. (1995). The size of the data-cube containing the source is $73^3 \sim 4 \times 10^5$ voxels. In the top-left panel we look along the frequency axis; in the top-right panel along the RA axis; and in the bottom the view is parallel to the geometrical major axis of the galaxy. For computing the projection we used an accumulate method. The different colors highlight different intensity levels in the data (i.e., grey, green, blue and red correspond to 3, 8, 15 and 20 times the rms noise respectively).
Chapter 2. The role of 3D interactive visualization in blind surveys of HI in galaxies

tail which is unsettled gas resulting from tidal interaction with another galaxy. The 3D structure of the HI data is, however, difficult to interpret for several reasons: i) the third axis of a data-cube is frequency, which is converted into a velocity applying the Doppler formula to the 21-cm HI line; ii) the measured velocities are the line-of-sight velocity components of a rotating system, therefore the 3D shape depends directly on the rotation curve; iii) in addition, the kinematic information of the gas is affected by geometric properties such as inclination, orientation of the semi-major axis, and gas distribution. Due to these complexities in the data, the user of a 3D inspection tool needs reasonable experience with the data and a certain learning period to assimilate the tool itself. This is not different from the situation 25 years ago, when radio astronomers had to train themselves to understand 2D visual representations such as movies of channel maps and position-velocity diagrams. During this learning process interactivity is a key factor (see 2.3.3 and 2.3.3).

The 3D visualization paradigm (volume rendering) described and used in this paper is limited by the use of 2D input and output hardware such as a standard monitor and mouse. A simple practical example of a limitation in 3D is the absence of a method for picking the value of one pixel with a cursor. Complementary visualization in 1-D and 2D can repair these deficiencies. Moreover, there is not a single best way to visualize a radio data-cube, but the combination of several methods (3D, 2D, 1-D, side by side, overlaid, blinking, etc.) and the interaction between them could deliver a very powerful analysis tool. It is important to view the data in different ways; this is the key to fully assimilating the information in the data. Therefore, a high-level of 1-D/2D/3D linked views must be achieved.

Very faint coherent signals, under $3\sigma$, are difficult to find even using 3D. Real-time smoothing can help in dealing with the noisy character of the data. In fact, if the signal is comparable to the noise, which will be the case for many Apertif observations, it is not possible to distinguish the signal itself from the noise at full resolution in any way. In Fig. 2.3, it is shown that only in the smoothed (60") version of the same data (in this case the signal-to-noise ratio of the filament is increased from $\sim 1$ to $\sim 4$) it is possible to localize a very faint filamentary structure that connects the two galaxies. It is already possible to detect the filament after applying a smoothing to a spatial resolution of 30" (signal-to-noise of $\sim 2$).

In the following use cases we will show how 3D interactive visualization helps in the analysis of the sources.
Figure 2.3 – Another view of the source in Fig. 2.2 is shown. The blue surface represents the full resolution data, while the green is the smoothed version at 60'' spatial resolution. Both surfaces are representations of the signal at 3σ. The green surface shows a very faint filamentary structure that connects the two galaxies.
Use Case A: analysis of sources with tidal tails

Fig. 2.4 explores the source shown in Figs 2.2 and 2.3 in more detail. A big tail due to a gravitational interaction is clearly present in the data-cube. It is very easy to recognize the tail in the volume rendering because the data are coherent in all three dimensions.

In the case of HI in galaxies one can extract additional information from fitting the observations with the so called tilted-ring model (Warner et al., 1973). Modeling tools (e.g., TiRiFic and 3D Barolo: Józsa et al., 2007; Di Teodoro and Fraternali, 2015c) generate a parametrized model data-cube, simulating the observed HI distribution of the galaxy as a set of concentric, but mutually inclined, rotating rings, which is then compared directly to the observation. This operation can give a deeper knowledge of the kinematics and morphology: asymmetries in surface density and velocity, presence of extra-planar gas, presence of inflows and outflows, gas at anomalous velocities, etc. However, these algorithms cannot recognize tidal tail structures and separate them from the central regularly rotating body of the galaxy. Combining 3D visualization with these algorithms through a 3D selection tool (e.g., Yu et al., 2012) will be very powerful. As shown in Fig. 2.4, separating the components visually enables a better view and a better understanding compared to the visual representation shown in the middle panel in Fig. 2.2.

A 3D selection tool will not only be useful for highlighting the different components with different colors, but also for retrieving quantitative
information (noise calculation, H I mass, velocity gradient, tilted-ring model-fitting, etc.) on the selected volume. For example, in the case of this PPScI source the user can separate the components and perform the calculations separately on the two volumetric selections. In this process, the key feature is the interactivity of the process itself.

**Use Case B: finding anomalous velocity gas**

It has been shown that the gas distribution of some spiral galaxies (e.g., NGC2403 shown in Fig. 2.5) is not composed of just a cold regular thin disk. Stellar winds and supernovae can produce extra-planar gas (e.g., galactic fountain (Bregman, 1980)). In this case, modelling can be used to constrain the 3D structure and kinematics of the extra-planar gas which is visible in the data as a faint kinematic component in addition to the disk. 3D visualization of both the data and the model can provide a powerful tool to investigate such features. The visualization tool could use the output model of automated model-fitting algorithms for visually highlighting the different components in the data-cube. In fact, if the model of the cold thin disk is subtracted from the data, it is possible immediately to locate any uncommon features in the data-cube of interest and have already an idea of their properties, directing further modeling. For example, a model of the extra-planar gas above or below the disk with a slower rotation and a vertical motion provides quantitative information about the rotation and the infall velocity of such gas.

In Fig. 2.5, the data of the NGC2403 observations are colored in green, while the blue structure is a tilted-ring model of regular rotation automatically fitted to the data with 3D Barolo. The top panel in Fig. 2.5 represents the position-velocity diagram along the semi-major axes which shows the typical rotation curve of a late-type galaxy plus some unsettled gas in the inner region. The middle panel is a 3D representation of the data, but it is very difficult to distinguish between the cold disk and the extra-planar gas. In fact, too much information is condensed in that visual representation. Separating and visually highlighting the different kinematic components, such as in the bottom panel, clearly shows the extra-planar gas. 3D visualization gives an immediate overview of the coherence. For example, it highlights the presence of extra-planar gas and its extension. On the other hand, for checking the data pixel by pixel it is better to use a two-dimensional representation like a position-velocity diagram.
Chapter 2. The role of 3D interactive visualization in blind surveys of H I in galaxies

Figure 2.5 – Three different illustrations of the H I data of NGC2403 from the THINGS survey (Walter et al. 2008b) are shown. The galaxy is very well resolved. The top panel represents the position-velocity diagram along the semi-major axes which shows the typical rotation curve of a late-type galaxy (the blue contours represent the model that fits the regular disk) plus some unsettled gas in the inner region (the lowest green contour of the data is at $3\sigma$). The middle and bottom panels illustrate two 3D representation of the data using an accumulate projection method.
2.4 Prerequisites for visualization of HI

Goodman (2012) has expressed that a visualization environment for astronomy should satisfy:

1. interactivity;

2. linked views with different representations of the data (2D, 3D and high-dimensional visualization);

3. availability of an open source repository and a high level of modularity in the source code for enabling collaborative work;

4. interoperability with Virtual Observatory (VO) tools through the Simple Application Messaging Protocol (SAMP; Taylor et al., 2011).

These requirements are also valid in our case, the visualization of HI in galaxies. Moreover, the interface must be able to handle astronomical world coordinates. This is of primary importance for many applications such as overlaying images taken at different wavelengths with other telescopes, cross-correlating source positions and velocities with existing catalogs, etc. A full overview of representation methodologies of celestial coordinates in FITS and related issues is given in Calabretta and Greisen (2002b) and Greisen et al. (2006a).

From section 2.3 we concluded that the data-cubes of interest will have dimension $< 10^7$ voxels ($< 0.25$ GB), but a large number ($\sim 100,000$) of small sources will be delivered by the surveys. Therefore, for quickly extracting the information from the data and presenting them in a clear and synthetic form, the visualization must be qualitative, quantitative, and comparative. In the next three paragraphs we will describe these demands and why we need three levels of visualization.

2.4.1 Qualitative visualization

First of all, astronomers want to look at their data in various ways in order to assess the data quality. An experienced astronomer can distinguish faint sources from the noise and instrumental artifacts, recognize the morphology and the kinematics of a galaxy, and identify unexpected HI emission (e.g., very faint structures such as extra-planar gas, tidal tails, and ram-pressure
filaments). Therefore, qualitative visualization will continue to play a major role.

In the previous section we showed the advantages and the drawbacks of adopting 3D visualization. Very fast interactivity in rendering, in 3D navigation, in data smoothing, and in quantitative and comparative functionality is important: if the interactivity is too slow, only the obvious signal will be found and subtle features may remain unnoticed. More precisely, the visualization should have a user-friendly interface capable to sustain navigation with more than 15 fps in order to provide the user with a fast interaction such as rotation, zooming, and panning of the data.

The interface should have the capability to change the transfer function (i.e., mapping the value of the projected voxels onto a color and transparency value) interactively to help the astronomer in the qualitative understanding of the data, both in the 2D and 3D visualization.

The user should also be able to choose different line-of-sight integrations during the process of projection for the volume rendering (e.g., minimum, maximum, accumulate). For example, in order to visualize H I absorption that is a negative line, a minimum transfer function is needed, while to see the H I emission in galaxies one can use a maximum or a very specific accumulate transfer function.

2.4.2 Quantitative visualization

Interactive quantitative visualization which allows the user to extract quantitative information directly from the visual presentation is of primary importance. In astronomy, and in particular in radioastronomy, this is not a new concept. For example the KARMA package is a very good quantitative tool in the framework of 1-D and 2D visualization. KARMA developers showed that a first level of quantification is to retrieve numbers from the visualized dataset and in some cases to represent them in a visual way for a better understanding. Examples are:

1. display of the flux value through a pixel in slice view and/or plot intensity profiles and display the value;

2. calculation of noise, standard deviation, maximum, minimum, H I mass or velocity gradient, etc., in a specific area or volume;

3. segmentation of the 3D data volume of an object;
4. construction and display of moment maps and position-velocity diagrams.

A second level of quantification can be introduced by having interactive features between the visualization and a plotting library (see, for example, the work in progress by Goodman (2012) and her team related to the GLUE Project[^5]. The idea is to plot quantitative information related to the data and then have a visual representation of that information in the visualization of the data. In order to give an idea of the benefits of this functionality, a hypothetical example follows: the first step is downloading HI, optical and infrared data, creating star formation rate (SFR) maps and plotting the local SFR values as a function of the HI column density ($N_{HI}$) of the pixel. The plot allows the identification of pixels deviating from the power law relation between SFR and $N_{HI}$. Subsequently, it will be possible to locate possibly deviant pixels by highlighting them in the 3D visualization. The second step is to examine where they are in the 3D data in order to assess whether they occupy specific regions, i.e., if they are coherent in the 3D data. The third step is retrieving quantitatively the SFR of a specific environment of the data-cube under investigation. For that it is necessary to select different zones using the visualization and then to plot the SFR/$N_{HI}$ of each zone with a different color (for example two regions in a spiral galaxy: the spiral arms and the bulge).

Standalone quantitative visualization is however not satisfactory. A synergy, using linked views, with comparative visualization is necessary for assessing the quality of the analysis, such as comparing a tilted-ring model with the data, and highlighting subtle faint structure in the data as we have shown in section 2.3.3.

### 2.4.3 Comparative visualization

In sections 2.3.3 and 2.3.3 we showed how in the case of HI in galaxies one can extract additional information from tilted-ring model-fitting.

The visualization tool should also enable an interactive comparison between data and models in order to check the quality of the model provided by the automated algorithm. This is possible by having the model routine embedded in the visualization interface. In fact, a coupling between model fitting and visualization will enable an interactive change of the parameters

Chapter 2. The role of 3D interactive visualization in blind surveys of H\textsc{i} in galaxies

of the model, such as rotation curve, column density, and inclinations as functions of the radius, and the comparison of the new model with the data. Interactive tilted-ring model fitting greatly helps in the analysis of warped galaxies. For example, Sparke et al. (2009) adopted an interactive procedure, using INSPECTOR, for arriving at the final model of NGC 3718 shown in the paper. INSPECTOR is an interactive tilted-ring modeling routine in GIPSY using a comparative visualization tool.

The comparison between an observation and a model of a galaxy can be made by examining 3D renderings of the data and the model in two separate windows, or by showing in one window an overlay of the model on the observation and in another window the difference between them. This separates regularly rotating gas from unusual kinematic features (extra planar gas, tidal tails, ram pressure induced structures). In addition, the interface needs to support display windows next to the 3D rendering with plots in which one can view results of the source analysis such as the rotation curve.

Comparative visualization can be also extended using models obtained by running $N$-body simulations (see Barnes and Hibbard 2009; Barnes 2011). This kind of systematic studies can benefit, in terms of speed and interactivity, from the usage of optimized $N$-body codes running on GPUs (Nyland et al. 2007; Portegies Zwart et al. 2007; Capuzzo-Dolcetta et al. 2013), some of which are publicly available via the Astronomical Multipurpose Software Environment (AMUSE; Pelupessy et al. 2013).

2.4.4 High-dimensional visualization techniques

High-dimensional data visualization (e.g., TOPCAT; Taylor 2005) of the parameter tables will enable the capability to have a full picture of the characteristics of the data in the catalog. This feature is very important to discover the unexpected. In fact, the catalog paradigm can fail if the number of sources is too large: in general it is possible to retrieve a list of data from catalogs using flags such as names or certain parameters of the objects; it is, however, usually not possible to have a general view of the main parameters of the sources in question. Therefore, a visualization package should be able to download tables that contain the required properties of galaxies (flux, flux error, degree of asymmetry, velocity width, integrated profile shape, etc.) and plot these parameters, allowing the user to find outliers. The user should also have the capability to mark the data
of interest in the plot and download the requested data-cube(s) from the catalog, using the interface for further exploration of the 3D signatures and comparing them with one or more models. This can be achieved using the SAMP protocol and other VO tools.

2.4.5 Summary

In this section we have defined the requirements that visualization of H I emission, in the survey era, must satisfy. We briefly summarize them here:

1. astronomical world coordinates in order to combine the visualization of H I data with data obtained at other wavebands;

2. 3D capabilities (i.e., presence of interactive volume rendering for grid data of dimension $< 10^7$ voxels and interactive color and opacity function widgets);

3. interactive linking between 1-D/2D/3D views;

4. quantification: physical data units, labels, and statistical tools;

5. linked 1-D/2D/3D selection tools;

6. 3D segmentation techniques;

7. interactive smoothing;

8. comparative visualization (multiple views, overlaid visualizations, etc.);

9. tools for generating tilted-ring and N-body models;

10. interoperability with VO tools.

2.5 Review of state-of-the-art 3D visualization packages

In the previous section we described in detail all the requirements a visualization tool must satisfy for enabling the source analysis that we outlined in section 2.3.3. A review of the current state-of-the-art of 3D visualization is very important in order to avoid duplication and
development of rendering algorithms and tools which may already exist. We performed a review of current 3D visualization software with the idea in mind that they have to satisfy the requirements listed in section 2.4.5 plus the following technical prerequisites. The software must:

1. run on multiple platforms;
2. have an intuitive interface;
3. have a Python wrapper for easy introduction of the SAMP protocol;
4. have a high level of modularity in the source code;
5. have proper documentation and long-term maintainability (i.e., presence of a significant user- and developer-community).

Many rendering algorithms and tools exist but we restricted the detailed review to a short list of publicly available, open-source and currently maintained packages with 3D interactive rendering capabilities:

1. **Paraview** ([Morelanda et al., 2007](#)): a general-purpose multi-platform data analysis and visualization application. The ParaView project started in 2000 as a collaborative effort between Kitware Inc. and Los Alamos National Laboratory.

2. **3DSlicer** ([Fedorov et al., 2012](#)): a software package for visualization and image analysis of medical data. It is natively designed to be available on multiple platforms.


4. **ImageVis3D** ([Fogal and Kruger, 2010](#)): a new volume rendering program developed by the NIH/NIGMS Center for Integrative Biomedical Computing (CIBC). The software is multi-platform and scalable.

For each package we performed a detailed review study in two steps:

1. a software user-friendliness survey: we tested the four packages by inspecting and analysing the H I emission of WEIN069 and NGC2403
We performed a survey by asking 15 radioastronomers to evaluate the intuitiveness and interactivity of the different features offered by each package using WEIN069 as a test data set. The evaluation involved each participant filling out a questionnaire after one hour of utilization of the packages. In all cases, the latest stable version of the software was used with the following hardware set-up: a Linux laptop (Ubuntu 14.04 LTS) equipped with an Intel i7 2.60 GHz CPU, an NVIDIA GeForce GTX860M GPU, 16 GB of DDR3 1.6GHz RAM, and a 15.6 inch monitor with a resolution of 1920 × 1080.

2. a source code evaluation: we performed a detailed study of the full source code, the level of modularity, and the available documentation for developers.

2.5.1 Review results

The resulting ranking of the packages is shown in Tables 2.1 and 2.2. In addition, we provide a detailed list of pros and cons for each package in Tables 2.3, 2.4, 2.5, and 2.6.

We can divide the packages into two classes: i) Paraview and 3DSlicer; ii) Mayavi2 and ImageVis3D. The software in the first group has many features, while the second group mainly offers qualitative visualization. The users noted that the interfaces offered by Paraview and 3DSlicer are complex, but at the same time, most of the users found Paraview rather intuitive. The intuitiveness (i.e., the learning time) ranking shown in Tables 2.1 and 2.2 obviously also depends on the experience of the users with similar visualization software.

The review highlighted that the users experienced a major lack of functionality in all four packages for: displaying labels with proper astronomical coordinates; 1-D visualization (e.g., line profiles); interactive smoothing; simple editing or blanking, and specific operations such as constructing a position-velocity diagram along a specified spatial axis; and comparative visualization (e.g., overlaid 1-D profiles and overlaid 2D contour plots on another image). This is not a surprising result. In fact, the packages considered in this section are aimed towards general or medical visualization purposes and lack the specialized visualization representations and interaction aspects common in radio astronomy. On the other hand, they do have advanced rendering capabilities, such as provided by the
Chapter 2. The role of 3D interactive visualization in blind surveys of H\textsc{i} in galaxies

<table>
<thead>
<tr>
<th></th>
<th>Paraview</th>
<th>3DSlicer</th>
<th>Mayavi2</th>
<th>ImageVis3D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>⊗</td>
<td>⊗</td>
<td>⊗</td>
<td>⊗</td>
</tr>
<tr>
<td>2</td>
<td>7%</td>
<td>27%</td>
<td>67%</td>
<td>0%</td>
</tr>
<tr>
<td>3</td>
<td>20%</td>
<td>73%</td>
<td>7%</td>
<td>0%</td>
</tr>
<tr>
<td>4</td>
<td>47%</td>
<td>53%</td>
<td>0%</td>
<td>7%</td>
</tr>
<tr>
<td>5</td>
<td>27%</td>
<td>60%</td>
<td>13%</td>
<td>7%</td>
</tr>
<tr>
<td>6</td>
<td>7%</td>
<td>20%</td>
<td>73%</td>
<td>13%</td>
</tr>
<tr>
<td>7</td>
<td>⊗</td>
<td>⊗</td>
<td>⊗</td>
<td>⊗</td>
</tr>
<tr>
<td>8</td>
<td>27%</td>
<td>53%</td>
<td>20%</td>
<td>33%</td>
</tr>
<tr>
<td>9</td>
<td>⊗</td>
<td>⊗</td>
<td>⊗</td>
<td>⊗</td>
</tr>
<tr>
<td>10</td>
<td>⊗</td>
<td>⊗</td>
<td>⊗</td>
<td>⊗</td>
</tr>
</tbody>
</table>

Legend: ⊗ = missing  ■ = not satisfactory  ■ = satisfactory  ▲ = good

Table 2.1 – A ranking of several 3D visualization packages is shown. The numbers in the first column refer to the summarized requirements in section 2.4.5. The colored bars are a representation of the ranking based on a user-test survey performed with 15 radioastronomers. Note that this software ranking is oriented towards the visualization of H\textsc{i} data (grid volume of dimension < 10\textsuperscript{7} voxels) in a desktop environment.
2.5. Review of state-of-the-art 3D visualization packages

<table>
<thead>
<tr>
<th></th>
<th>Paraview</th>
<th>3DSlicer</th>
<th>Mayavi2</th>
<th>ImageVis3D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>![](13%</td>
<td>47%</td>
<td>40%)</td>
<td>67%</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend: ⊗ = missing  ■ = not satisfactory  ◻ = satisfactory  □ = good

Table 2.2 – A ranking of several 3D visualization packages is shown. The numbers in the first column refer to the technical prerequisites listed in this section 2.5. The colored bars are a representation of the ranking based on a user-test survey performed with 15 radioastronomers. Note that this software ranking is oriented towards the visualization of H1 data (grid volume of dimension < $10^7$ voxels) in a desktop environment.

Visualization Toolkit (VTK), and a modern, multiple-platform, reliable interface based on Qt. For example, the packages enable the user to save the whole working session in a bundle: the data, the visualization, and the module structure used for the analysis.

At the moment, all the packages listed lack multi-volume rendering. Multi-volume rendering is the operation to render two or more volumes on the same space. This feature is necessary for enabling very fast 3D overlaid comparative visualization.

2.5.2 Visualization of H1 and 3DSlicer

In the previous section, we showed that no single package of the four investigated meets all of the requirements. A highly appropriate decision to use 3DSlicer was made based on the priorities and requirements, with a clear list of functionality for development. Despite the complexity of the interface, we chose to adopt 3DSlicer as base platform for the development.

---

*http://www.vtk.org/
*http://qt-project.org/*
Chapter 2. The role of 3D interactive visualization in blind surveys of H I in galaxies

Paraview

Pros

• CPU/GPU rendering based on the Visualization Toolkit (VTK);
• skill to connect to a server to do the computation;
• editable interface with unlimited 2D/3D views;
• linked 1-D/2D/3D views;
• cropping and selection tools;
• 3D segmentation techniques, i.e., isosurfaces;
• skill to perform statistics on the user selection;
• high level of modularity in the source code;
• embedded python console in the interface for fast interaction with the source code;
• presence of documentation both for users and developers.

Cons

• the interface is complex;
• astronomical world coordinates and labels not displayable;
• the interface is not optimized for 1-D and 2D visualization;
• interactive smoothing missing.

Table 2.3 – A list of pro’s and con’s relative to the Paraview package is presented. The advantages and disadvantages listed form a detailed description of the feedback provided by the authors and the users of the software survey shown in Tab.2.1 and 2.2.
### 3DSlicer

**Pros**

- CPU/GPU rendering based on VTK;
- interface is also optimized for 2D visualization of channel maps;
- high-level of linking between 2D and 3D views;
- interactive cropping and selection editor tools;
- skill to perform statistics on the user selection;
- 3D segmentation techniques, i.e., isosurfaces;
- high level of modularity in the source code;
- embedded python console in the interface for fast interaction with the source code;
- presence of documentation both for users and developers.

**Cons**

- the interface is very complex and not intuitive;
- astronomical world coordinates and labels not displayable;
- 1-D visualization missing;
- 2D contour plots missing;
- interactive smoothing missing.

**Table 2.4** – A list of pro’s and con’s relative to the 3DSlicer package is presented. The advantages and disadvantages listed form a detailed description of the feedback provided by the authors and the users of the software survey shown in Tab.2.1 and 2.2.
Mayavi2

Pros

- CPU rendering based on TVTK (Python wrapper for VTK);
- cropping and selection tools;
- 3D segmentation techniques, i.e., isosurfaces;
- contour plots;
- a simple and clean scripting interface in Python, easy integration with other python libraries.

Cons

- the interface is not stable;
- presence of only CPU rendering capabilities. The frame rate per second is low, $fps < 5$, for data-cubes bigger than $10^6$ voxels;
- color transfer function widget is not easy to use;
- astronomical world coordinates and labels not displayable;
- 1-D visualization missing;
- interactive smoothing missing;
- lack of statistics tools.

Table 2.5 – A list of pro’s and con’s relative to the Mayavi2 package is presented. The advantages and disadvantages listed form a detailed description of the feedback provided by the authors and the users of the software survey shown in Tab. 2.1 and 2.2.
### ImageVis3D

**Pros**

- very light, fast, and intuitive interface;
- GPU rendering;
- 3D segmentation techniques, i.e., isosurfaces.

**Cons**

- the long-term maintainability of the rendering code is unknown;
- astronomical world coordinates and labels not displayable;
- 1-D and 2D visualization missing;
- interactive smoothing missing;
- lack of statistics tools;
- lack of documentation.

*Table 2.6* – A list of pro’s and con’s relative to the *ImageVis3D* package is presented. The advantages and disadvantages listed form a detailed description of the feedback provided by the authors and the users of the software survey shown in Tab.2.1 and 2.2.
of a HI visualization tool. Our choice has been the result of considering various factors such as the presence of adequate documentation, the number of people actively working on the software, and quantitative features already implemented in the interface. These three main factors make 3DSlicer the best solution for us. In fact, the medical visualization needs are indeed very close to the astronomical ones. For example, the interface layout and the navigation through the data are already optimized for parallel 2D visualizations (e.g., movies of channel maps). The following features need to be added to 3DSlicer in order to fulfill the requirements described in section 2.4:

1. proper visualization of astronomical data-cubes using the data formats FITS, HDF5, CASA, and Miriad;
2. enabling interactive smoothing in all three dimensions and multi-scale analysis, such as wavelet lifting;
3. generation of flux density profiles, moment maps and position-velocity diagrams linked with the 3D view;
4. interactive 3D selection of HI sources;
5. interactive HI data modeling coupled to visualization;
6. introduction of the SAMP protocol to enable interoperability with Topcat, and other VO tools and catalogs.

2.6 Concluding Remarks

HI observations are moving into the era of big surveys. Upcoming HI surveys, such as those envisaged with Apertif and ASKAP, will deliver big datasets leading the radio astronomer into the regime of the so-called *Fourth Paradigm* (i.e., data-intensive scientific discovery; Hey et al. 2009).

Apertif is expected to start its observing campaign of the northern sky in 2017. The daily Apertif data-cube will have dimensions of $2048 \times 2048 \times 16384 \sim 68$ Megavoxels and the expected number of HI source detections is $\sim 100$ every day. WALLABY will have similar characteristics. The large volume of data creates new needs, in terms of tools and algorithms which must exploit new ideas and solutions for storage, data reduction, visualization, and analysis to obtain scientific results.
Visual analytics, the combination of automated data processing with human reasoning, creativity and intuition, supported by interactive visualization, is one of the prime methodologies that allow keeping the human in the investigation loop. In this paper, we defined the visualization prerequisites and future perspective for applying this paradigm to HI observations focusing on the introduction of 3D visualization in the process of source finding and analysis. In fact, the current astronomy visualization software has very limited 3D capabilities for grid data (section 2.2); while general purpose visualization software (section 2.4, 2.5) is not aimed at the analysis of HI data.

In this paper we showed:

1. more than 99% of the voxels in the HI datasets that Apertif will deliver is dominated by noise and the sources are hidden in it (see Fig. 2.1). The current source finder software can extract them with high reliability and completeness (Whiting 2012; Serra et al. 2015). The typical volume of individual sources will be $50^3 = 1.25 \times 10^5$ voxels (up to $512^3 \sim 1.3 \times 10^8$ in the case of occasional large galaxies), reducing the storage, I/O bandwidth and computational demands for visualization to a level accessible on desktops and laptops. The predicted weekly data rate, on the other hand, is high ($\sim 10^3$ sources). Fortunately, only a subset of these (2-3 sources per day) will be highly resolved (more than 10 resolution elements) or show complex features such as tails and extra-planar-gas. A powerful interactive visualization tool will be needed for fast inspection and analysis of these objects.

2. the analysis of the sources, for example producing moment maps and rotation curves, will also be done in an automated way. In particularly complex cases, human interaction will be necessary to drive the automated algorithm in the data volume and provide immediate feedback on the quality of the results (see section 2.3.3). Visualization tools with supervised semi-automated analysis algorithms will be needed. In fact, it becomes necessary to produce refined data with minimal time, but maintain the same level of quality. For example, the derivation of the rotation curve of a galaxy passes through the creation of the so-called tilted-ring model which, then, is compared to the data. This process has been converted to an automatic algorithm. However, significant kinematic features different from the Keplerian
rotation (e.g., tidal tails, see Fig. 2.2) will be present in part of the data. The current algorithms cannot automatically flag these features for the analysis. Therefore, human intervention is necessary to separate the regularly rotation disk and different kinematic features, and to feed the fitting algorithm with the selection, so that the user can judge the results quantitatively.

3. In section 2.3.3, we showed that 3D visualization can enable an immediate overview of the kinematics of a galaxy, leading to improved understanding of the coherence in the data. Moreover, a high level of interactivity in all visualization aspects (rendering, smoothing, retrieving quantitative information, and comparative features) will be the key for enabling a fast inspection of the data. On the other hand, volume rendering has its limitations due to current 2D input and output hardware. Some examples of these limitations are projection issues and the impossibility to move the cursor pixel by pixel. Adding 1-D/2D views linked to the 3D representation resolves these limitations. The combination with high-dimensional visualization techniques, which can help in finding outliers and patterns in the oceans of data, is also necessary.

4. In section 2.4, we identified the requirements for the visualization and analysis of H I in galaxies: interactive visualization with quantitative and comparative capabilities with 3D selection techniques and supervised semi-automated analysis. Moreover, the source code must have the following characteristics for enabling collaborative work: open, modular, well documented, and well maintained. After a study of the state-of-the-art of the open-source and actively maintained visualization packages with rendering of grid data capabilities (see section 2.5), we adopted 3DSlicer as a platform for developing a fully interactive desktop H I data visualization tool with quantitative and comparative features (section 2.5.2). These techniques can also be used for other astronomical datasets such as 3D datasets provided by recent Integral Field Unit (IFU) observations (Sanchez et al., 2012; Karman et al., 2014; Richard et al., 2015). In that case, collaborative work is necessary to identify the key features needed to provide quantitative visualization.
In conclusion, the success of a visualization tool depends heavily on the number of people using it over its life time. The life time of a software package depends on several factors such as usability, maintainability, and whether it has been developed with good insight in the subtle aspects of the data and its interpretation. **KARMA** is a perfect example of a successful package, developed in the mid 1990’s but still widely used by radio astronomers to date. Our aim is to achieve an analogous result exploiting the current hardware and algorithmic paradigms, focusing on the linking between 2D and 3D visualization, quantitative/comparative features and high-dimensional visualization.
2.7 Additional on-line material

In this section we provide videos of the volume rendering of part of the data presented in this paper.

![Volume rendering of the data](https://www.youtube.com/watch?v=sS_5LrOS5bo)

![Volume rendering of the data](https://www.youtube.com/watch?v=yLjW9nbO8g)

Figure 2.6 – Volume rendering of the data of the top-right panel of Fig. 2.1 (left panel) and of Fig. 2.2 (right panel) are shown.

2.8 Acknowledgments

Two of the authors, D. Punzo and J.M van der Hulst, acknowledge support from the European Research Council under the European Union’s Seventh Framework Programme (FP/2007-2013) / ERC Grant Agreement nr. 291531. We thank R. Sancisi and F. Fraternali for very useful feedback. We also thank E. di Teodoro for providing 3D Barolo to us. Finally, we thank the reviewers for their constructive comments, which helped us to improve the paper substantially. Figures and videos in this paper were generated by using 3DSlicer [http://www.slicer.org/].