Imaging Pipeline Software

Bobbie Ware 17962233

Math705 Research Project

Abstract

Aperture synthesis is the process of taking data from interferometry telescope arrays and produces an image of the sky. This project covers the gathering of knowledge on the required topic to create software capable of carrying out this process. The software will be tested against ideal models and changed and improved upon based on those results.

Please not this was submitted for a mid-project review and so the report is not finished

Introduction

An imaging pipeline for Image Synthesis is designed to take data either gathered by radio interferometry telescopes, or generated to simulate those telescopes, and form an image of an area of the sky. This software will be similar to that which will be used for the Square Kilometer Array, a project that AUT is involved in, where the same processes will have to take place.

This project will follow a Design Science methodology where software will be the generated artefact used to experimentally investigate image synthesis. The project will start with the gathering of knowledge on the techniques used in an imaging pipeline. It will also involve gaining knowledge on solutions to complications such as concurrency control and the mapping of visibilities to a grid. Then the project will involve implementing the techniques in the form of a Java program and then will be tested using visibility input data available in the High Performance Computing Research Laboratory. The software will be testing against other pipelines and changes will be made to try to improve its performance.

The output from the developed software, using visibility data as input, will be an image of the sky. The images produced by the pipeline will be analyzed to compare it against the known sky images for the data sets to validate whether the techniques are implemented properly and potentially look at its performance.

It is expected that an imaging pipeline will be developed with the capability for image synthesis. Also expected to gain knowledge in the three main steps involved in the pipeline, namely gridding, (inverse) Fourier transform, and deconvolution, as well as some techniques for algorithm optimization.

Table of Contents

[Introduction 2](#_Toc5715225)

[Literature Review 4](#_Toc5715226)

[Synthesis Imaging 4](#_Toc5715227)

[Fast Fourier Transform (FFT) 5](#_Toc5715228)

[Gridding 5](#_Toc5715229)

[Deconvolution 6](#_Toc5715230)

[Methods 7](#_Toc5715231)

[Results 8](#_Toc5715232)

[Discussion of Results 9](#_Toc5715233)

[Conclusion 10](#_Toc5715234)

[**References** 11](#_Toc5715235)

[Appendix 12](#_Toc5715236)

Literature Review

## Synthesis Imaging

The resolution of radio telescopes can be increased by using pairs of telescopes (baselines) and taking the product of the received signals. This resolution can be changed by increases the separation of the baseline, rather then increasing the size of the individual telescopes. This method popularized by the work of (Ryle & Hewish, 1960) states that using these baselines it produces “exactly the same result as that obtained by using the complete large aperture”. This technique allowed for cheaper production of much larger apertures and the eventual development of the techniques used now.

These techniques gather Fourier domain data in the form of a visibility, however the way in which they are sampled is non-uniform, so we must place it on a rectangular grid. This process is known as gridding and the methods used now are based on the work by (Brouw, 1975). These visibilities V(u, v) fall upon the plane in which the baselines are setup, for a wider coverage of this place more baselines can be added and could also be moved around.

With more modern telescopes being developed, moving them around became a substantial task and instead the rotation of the earth can be used to move these points around the plane. An image of these points on the V(u, v) plane can be seen on the left of Figure 1.

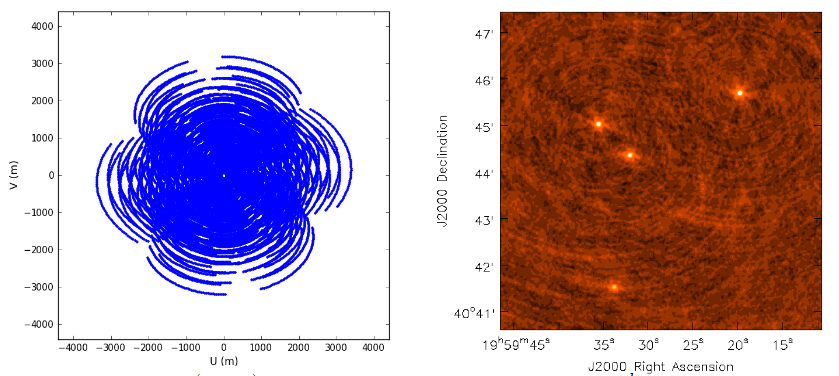
As there are gaps within the V(u, v) plane the image is a “dirty image”, this can be seen on the right of Figure 1.

Figure 1. V(u,v) plane showing data points. (Rau, 2012) Retrieved from Australian Telescope National Facility, from https://www.atnf.csiro.au/research/radio- school/2012/lectures/tue/RVU\_ImagingDeconvolution.pdf

## Fast Fourier Transform (FFT)

A Fourier Transform is a process for signal-processing and analysis. (Brigham, 1988) states that the extent of the use the process is as follows “biomedical engineering, imaging, analysis of stock market data, spectroscopy, metallurgical analysis, nonlinear systems analysis, mechanical analysis, geophysical analysis, simulation, music synthesis”. It is widely regarded as one of the most important algorithms based on its impact in so many areas. Simply put a Fourier Transform is used to show different parts of a continuous signal, however for Interferometry an Inverse FFT (iFFT) is used as we are combining the amplitude and phase of the signal to form an image. The inverse Fourier transform is used as we are taking the data from the Fourier Plane. Using such a method is based upon the work of (Hogg, MacDonald, Conway, & Wade, 1969). The algorithm used was first discovered by Gauss and later rediscovered by (Cooley & Tukey, 1965) which also notes that “Wherever possible the use of N = r^m with r = 2 or 4 offers important advantages” which impacts the design of the pipeline.

## Gridding

Gridding is the process of mapping the data collected onto a rectangular grid so that it may be processed by the iFFT and then displayed as an image. Early techniques for places the visibility data on a grid involved placing the visibilities upon the closest grid point that aligns with the plane on which they were gathered and either adding then all together or averaging them out. Early methods where used by (Hogg, MacDonald, Conway, & Wade, 1969). However, this led to artifacts forming and therefore a limited application for the process. An alternative method was first used by (Brouw, 1975) and would take a weighted value based the distance between local grid point and the point of the visibility. By designating a “support” area around the local grid point the data can be added to these areas. An ideal gridding method was given by (O'Sullivan, 1985) with his gridding algorithm that used a sinc function was “computationally efficient” and resulted in “arbitrarily small artifact levels”, however this function would give infinite extent to the support. This is not ideal computationally for the gridder so instead the use of convolution kernels with a set support are commonly used.

A simplified approach to gridding follows these steps. For every visibility, find the closest grid point to the data on the V(u, v) plane, using the convolution kernel the data point is spread across the support region.

## Deconvolution

Once an image is formed from the Fourier Transform is it called a ‘dirty image’, as seen in Figure 1, this is due to the effects of having limited sampling of the V(u, v) plane. The process of Deconvolution can be used to ‘CLEAN’ the image. This method by (Högbom, 1974) uses the original V(u, v) data to form a “dirty beam”, then by taking away the dirty beam from the points of the dirty image with the greatest brightness you are left with a residual image. By iteratively doing this process a clean image is left behind.

This was expanded upon by (Clark, 1980) to make it more efficient. His method involves using more FFT’s in a major and minor cycle to subtract points away from the dirty image.

Further examples of the ‘CLEAN’ algorithm and the Maximum Entropy technique by (Skilling & Bryan, 1984) are compared by (Cornwell & Bridle, 1996).

Methods

## Design of pipeline

The pipeline will consist of three main sections, the Gridder, the inverse Fourier Transform, and the Deconvolution. Before that I will break down some design choices made before coding began.

For the language to be used I decided on Java, this is due to it being the language I am most familiar with. Java is the language primarily used in my computer science papers at AUT. It is also more then capable of handling the size of the data I will be using and supports parallelization.

The grid size to be used will be 1024x1024. This us due to wanting a high enough resolution to be able to properly test against, and the grid length and height must be a result of 2^x due to the (Cooley & Tukey, 1965) radix 2 FFT we are using. This algorithm has a much better run time then a standard Fourier Transform under the condition that the data is ordered, hence why we grid the visibilities.

To test my pipeline, I will use a dataset generated by the HPCRL at AUT. By using this data I can better check how well the Fourier Transform and deconvolution is working as it comes with a perfect image made from the real components of the sky. This perfect image comes from the data being generated, with captured data we never know the true sky image and so it is les useful to compare. As well as a perfect image the data contains Visibilities.csv, a file containing around 23000 data points. Each visibility consists of its locations along the (u, v) plane, following by its value as a complex number. The dataset also includes the Prolate Spheroidal, used as the convolution kernel for the data with a support size of 7, however for better precision it has been oversampled by a factor of 4. The last thing needed from the data is the configuration for achieving the best accuracy for the gridder. In order to correctly place the visibilities two thing must be used, a wavelength to meters ratio, and the UV scale. For the data generates a frequency of 300000000 was used. To calculate the scale we use the product of the grid size and the cell size, our grid size is 1024 as discussed earlier, cell size is 4.848136811095360e-06 as given by the data. We used these values as follows.

uLocationOnGrid = (-uCoordinate \* wavelengthsToMeters) \* UVScale;

vLocationOnGrid = (vCoordinate \* wavelengthsToMeters) \* UVScale;

A double is used as the primitive data type as in the Java language (Oracle, 1993) is able to store values from 4.94065645841246544e-324 to 1.79769313486231570e+308 including values with high decimal accuracy needed.

## Gridder

Implementing the gridder begins with initializes two 2d double arrays, these are used to store the real and imaginary values respectively, as the values are placed on the grid they will be stored in these. Following that the gridder loads the data from the cvs file and storing it as a double array. The first two indexes are the u and v coordinates, after that is the real and then imaginary values. These double arrays are placed in linked list data type to ensure the size of the list is not an issue. Then going through the list, each array is transformed as per the formula mentioned before.

Once the visibilities are modified correctly the process of placing the points on the grid begins. Firstly, we take the location modified by the UVScale and round it to the nearest integer to find the closest grid point. The using the support value of 7 we take a 7x7 grid around this center grid point as where we will place the data point. Using our convolution kernel we form a 2d array 27x27, due to our x4 oversampling, and finding the distance of the grid point to the true visibility point

Results

Discussion of Results

Conclusion

**References**

Brigham, R. O. (1988). *The Fast Fourier Transform and its Applications.* Prentice-Hall.

Brouw, W. N. (1975). Aperture Synthesis. In C. De Jager, & H. Nieuwenhuijzen, *Image Processing Techinques in Astronomy* (pp. 301-307). Dordrecht: Springer.

Clark, B. G. (1980). An efficient implementation of the algorithm 'CLEAN'. *Astronomy and Astrophysics*, 377-378.

Cooley, J., & Tukey, J. (1965). An algorithm for the machine calculation of complex Fourier series. *Mathematics of Computation*, 297-301.

Cornwell, T., & Bridle, A. (1996). *Deconvolution Tutorial*. Retrieved from National Radio: https://www.cv.nrao.edu/~abridle/deconvol/deconvol.html

Högbom, J. (1974). *Astronomy and Astrophysics Supplement Series*, 417.

Hogg, D. E., MacDonald, G. H., Conway, R. G., & Wade, C. M. (1969). Synthesis of Brightness Distribution in Radio Sources. *Astronomical Journal*, 1206-1213.

Oracle. (1993). *Class Double*. Retrieved from Javadocs: https://docs.oracle.com/javase/7/docs/api/java/lang/Double.html

O'Sullivan, J. D. (1985). A Fast Sinc Function Gridding Algorithm for Fourier Inversion in Computer Tomography. *IEEE Transactions on Medical Imaging*, 200-207.

Rau, U. (2012, Sept 24). *Imaging and Deconvolution.* Retrieved from Australia Telescope National Facility: https://www.atnf.csiro.au/research/radio-school/2012/lectures/tue/RVU\_ImagingDeconvolution.pdf

Ryle, M., & Hewish, A. (1960). The synthesis of large radio telescopes. *Monthly Notices of the Royal Astronomical Society, Vol. 120*, 220-230.

Skilling, J., & Bryan, R. K. (1984). Maximum Entropy Image Reconstruction. *Monthly Notices of the Royal Astronomical Society*, 111-124.

Appendix