RIDIC: Real-time Image Display Component for the *Jürgen Stock* Telescope of Venezuelan National Astronomical Observatory

Iván Vivas (ievivas@cida.ve), Andrés Arcia-Moret (andres.arcia@ula.ve),
Domingo Hernández (hdd@ula.ve), Carmen Rodríguez (rodriguez@cida.ve)

Abstract — **RIDIC displays in real time images captured by a digital electronic camera installed at the Jürgen Stock Telescope of Venezuelan National Astronomical Observatory (NAO). It is designed with a client-server architecture, in which the Server (RIDIC) receives from Client the numeric values that represent the light intensity of a sky image. Data transfer is done through sockets using the TCP transport protocol. RIDIC captures data and then constructs, processes and displays the digital image through its Graphical User Interface (GUI). RIDIC was developed in C++, using elements provided by the Qt library.**

***Keywords* — Astronomical Image Displaying, Digital Image Processing, Performance Analysis, TCP / IP.**

I. Introduction

CIDA (*Centro de Investigaciones de Astronomía:* Research Center for Astronomy) is the most important institution in Venezuela for astronomical research. CIDA is responsible for managing the NAO, located near Merida city (Venezuela). NAO has four telescopes: a double astrograph telescope, a reflector telescope, a refractor telescope and a Schmidt telescope, also named Jürgen Stock telescope (the most important of the four). The later has a digital electronic CCD camera (QUEST[1] camera), installed in 1998.

The QUEST camera system has electronic and software elements that control the acquisition, storage and processing of astronomical digital images. Through a GUI called Control Panel or "Observe Program", it handled the necessary parameters to make an observation. RIDIC is a complement to this program, which allows the image display of the 16 CCD sensors (one sensor at a time) of the QUEST Camera in drift scan[2] mode observations. RIDIC is designed with a client-server architecture, the client (RIDIC) receives integer values representing the light intensity of a row of pixels of a sky image. This transfer is done reliably and orderly through Sockets, using TCP transport protocol that is part of the protocol stack of the QNX operating system, used by the server program. RIDIC captures data lines and then constructs, processes and displays a row of a digital image through its GUI. Digital image processing is necessary; because astronomical images have high contrast. Therefore, this should be enhanced, improving the visual appearance, that allows to visualize faint stars. The contrast enhancement technique used in RIDIC is a modification of Linear Stretch Method [1].

In the system, real-time displaying and adaptation to sky conditions, allow a preliminary visual evaluation of astronomical images quality, minimizing the observer's reaction time to solve problems during an observation. The observation time in clear nights is critical, since in NAO over tan 60% of nights of the year are cloudy; hence, the number of useful images captured during a clear night should be maximized. In addition, RIDIC creates an ideal work environment in an astronomical observatory, since the user can see on screen the piece of the sky that the telescope is capturing at the moment.

II. Theoretical Bases

A. QUEST Camera subsystems

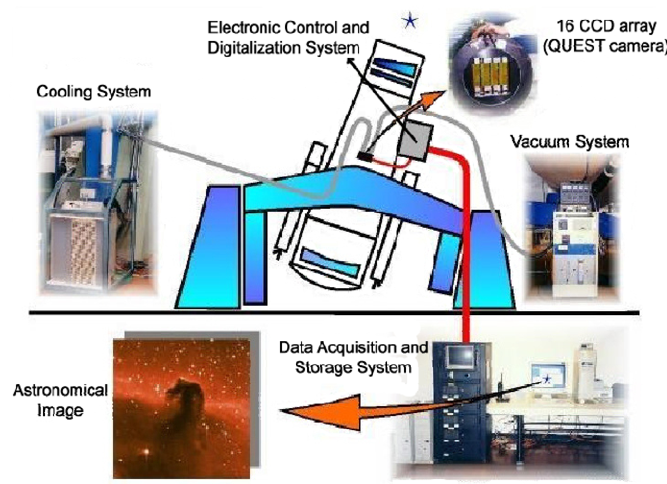

FIG. 1: QUEST Camera System

The 67.1 Megapixel QUEST camera mainly consists of a 16 CCD array. However, the complete camera system consists of a set of subsystems that are briefly described below:

II.A.1. CCD Sensors: QUEST camera containing a set of 16 CCDs of 2048 x 2048 pixel-sensors[3] of 15 µm, arranged in columns or "moving fingers" with four sensors each. This

[1] QUasar Ecuatorial Survey Team.
[2] One of the two observations modes with the Stock telescope.

[3] Photosensitive elements arranged in a grid. Falling photons (light) on them, release electrons which are quantified (photoelectric effect).

mobility allows fingers to adapt to driftscanning at different declinations[4]. Sensors are installed in the focal zone of the telescope (see Fig 1). A set of filters can be mounted on the detectors, which can block (filter) certain wavelengths, leaving only those needed to study.

II.A.2. Electronic Control and Digitalization (ECD): It consists of a set of 32 electronic cards, of which 16 are digital cards and the other 16 are analog / digital. Cards are installed in a cube-shaped chassis located outside of the telescope (see Fig 1). These cards are used to synchronize and read the 16 detectors of QUEST camera. Each CCD is controlled by two cards: one digital and one analog / digital. The digital card is the responsible for generating and distributing timing signals necessaries for CCD's operation. The analog / digital card is responsible for receiving the analog signal from the CCD's, to process it and send it to the data acquisition system.

II.A.3. Cooling and Vacuum: These subsystems are located near the telescope (see Fig. 1) and reduce thermal noise[5] that can saturate an image in seconds. At higher temperatures, higher thermal noise and vice versa. Therefore, to reduce thermal noise is necessary to cool the entire set of CCDs to very low temperatures. This is accomplished through a closed loop cryogenic system [2], which can reach temperatures under -80 °C. However, as the temperature decreases to low values, the air humidity condenses on cold surfaces, so to avoid condensation problems, the lines that carry the coolant to the vacuum chamber are vacuum.

II.A.4. Data Acquisition and Storage (DAS): This subsystem consists of a computer network that describes an inverted tree topology in a 4-2-1 configuration. Its current distribution consists of seven computers, four of which are for data acquisition, two for data storage and one for the user interface. This subsystem is described in more detail in section III.

B. Observation modes

Before describing the observation modes, it is important to describe how the stars move in the sky, because the telescope and the QUEST camera must adapt to this movement to make a proper astronomical observation.

Sidereal motion: It refers to the star improper movement produced by Earth rotation. This apparent shift, describes a different path when the star is farthest from the celestial Equator[6]. The stars on the celestial Equator describe straight east-west trajectories; however as the stars are located furthest from the celestial Equator, their trajectories describe arcs whose radius decreased while the stars are closer to celestial poles. Just at the poles, the apparent shift of the star would be zero.

The two modes of observations of the QUEST camera are:

II.B.1. Guided mode: It is the traditional technique of observation. It consists in pointing the telescope at a sky sector and follow their sidereal motion. This is achieved by an electric motor that drives the rotational movement of the telescope. In this mode the camera shutter opens and stays there for a predetermined time. Once the exposure time elapsed, the shutter closes and the still image of the sky that was formed on the CCD is transferred to the computer. That is, the CCD is read after an exposure. In this mode it is recommended to move the QUEST camera fingers to a parallel position. A guided mode observation creates an image for each CCD detector. Thus, 16 digital images are obtained.

II.B.2. Drift scan mode: Unlike guided mode, in drift scan mode the telescope remains oriented to a fixed position and the sky "moves" over it. The shutter will remain open for the observation. A single observation in this mode can last all night, because the detectors are read continuously, but for every 2048 rows, it generates a digital image. RIDIC is designed to work in this mode.

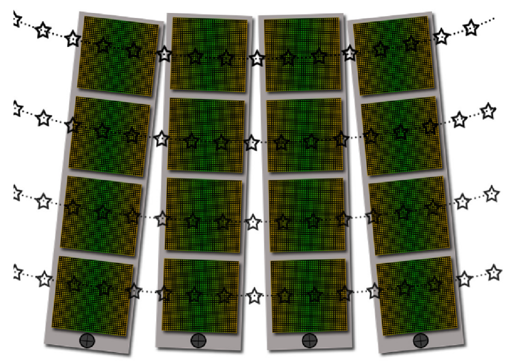

FIG. 2: Moving fingers to adapt to the sidereal motion

Observed objects must preserve their real shape, so the observation must comply with two conditions simultaneously [3]:

i. Stars trajectories should be straight and occur, on average, along one line of CCD pixels-sensors. For this reason, the fingers should be moved to suit the best for star sidereal motion (see Fig. 2).

ii. Reading the rows of CCD pixel-sensors should be performed at a synchronized frequency with the star linear velocity. That is, the reading frequency must be a function of the star sidereal motion and telescope declination. On the Equator this frequency is 14.5 Hz.

C. Data Acquisition Process

The process begins with selecting the observation mode and other parameters through the Observe Program. Generally, for any of the two observation modes, the image acquisition begins with the opening of the camera shutter, a device that allows the entry of light to the CCDs. Then, the electric charge collection phase starts, in which photons (light) interact with the silicon atoms (CCD components) and release electrons (photoelectric effect). These electrons are collected into packets that travel one by one to an amplifier, which then handles the process of quantification. This process measures the charge of each packet and generates a proportional voltage

[4] Declination (δ) and right ascension (α) are coordinates used by astronomers to locate objects in the sky.

[5] CCDs not just release electrons by receiving starlight, but so does it by the effect of temperature.

[6] Projection of Earth Equator in space.

to the amount of light received by each pixel-sensor. The process of moving the charged packages must be repeated again and again until reading the entire CCD pixel-sensors array.

CCD output signal is the voltage generated by the amplifier. This analog signal is converted to digital signal through the analog / digital card that is part of the ECD (see blue connection in Fig. 3). Then, the digital signal is transferred to the Data Acquisition subsystem. This subsystem is composed of four computers (PC1, PC2, PC3 and PC4 in Fig 3), where each one receives digital signals from a QUEST camera finger. That is, each computer receives data from four CCD sensors. These computers built digital images using FITS file format. Because the computers have limited storage capacity, data is transferred to other two computers with more disk space, part of the Data Storage subsystem (PC5 and PC6 in Fig. 3). Data in PC1 and PC2 is stored in PC5 and data in PC3 and PC4 is stored in PC6. Computers PC1-PC6 use QNX operating system and have not change since its installation in 1998; they have the same hardware and software since then, and they only differ in their functions within the system.

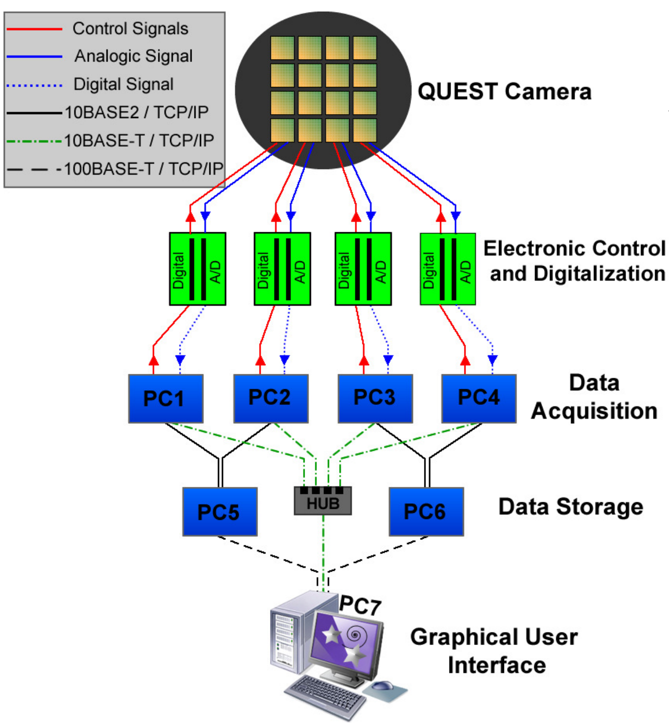

FIG. 3: Connection Diagram of QUEST Camera System [7]

Finally, to see the astronomical image, it must be transfered to PC7. This computer uses Linux as operating system (Fedora Core 7 distribution) and was last updated on September 2007. PC7 have the Observe program installed and other programs for image quality analysis. Additionally, at the end of each observation night, data (astronomical digital images) are transferred to PC7 and to an external hard drive, which is used to move data to CIDA facilities in Mérida city for further use in scientific research.

D. Digital Image generated by QUEST Camera

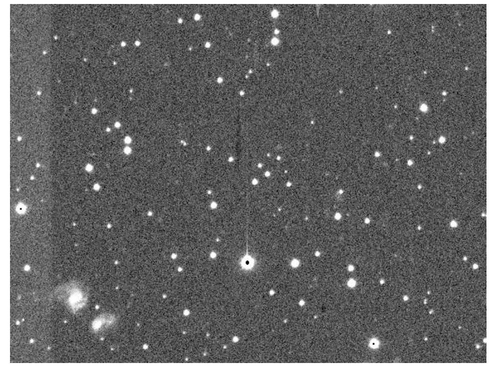

FIG. 4: Astronomical Digital Image generated by a single CCD of QUEST Camera

The QUEST camera generates digital images (see Figure 4) with the following characteristics:

- Raster image or bitmap.
- Grayscale.
- 2100 x 2048 pixels (8,6 MB).
- 16 bpp color depth of 65.536 gray levels.
- Color depth is represented in direct color. Thus, a color table is not used but instead each value is stored directly in the pixel.
- The file format is FITS, which is an uncompressed format.

III. ARCHITECTURE AND COMMUNICATION OF THE QUEST CAMERA SYSTEM

A. Software Architecture of the QUEST Camera System:

Fig. 5 shows the system software modules. It also shows the interactions between them, the protocols used to exchange information, where the memory mapping is and the input / output devices calls. This architecture presents 10 modules [4] that bring together the necessary programs aimed to reach the science goals of the QUEST project. These modules are distributed in the seven computers connected by Ethernet network, shown in Figure 3.

[7] Note that for clarity, the diagram only includes 4 CCD connections, but really each CCD is connected to two cards for signal control and digitization.

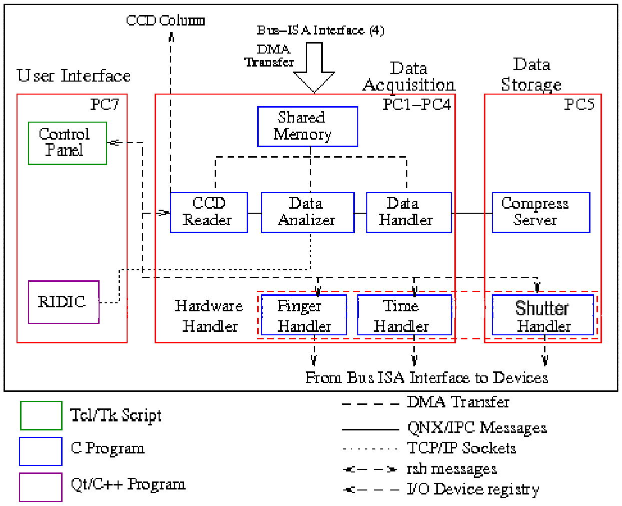

FIG. 5: Software Architecture of QUEST Camera System

Initially, data transfer is done from the ECD to the Data Acquisition subsystem, via a coaxial cable to the CCD controller cards that are built into PC1-PC4. These cards have an ISA bus that performs data transfers via Direct Memory Access (DMA). Each of these buses access memory to read or write independently from the main processor.

The CCD Reader module starts the detector control and Analyzer/Data Handlers modules. The Analyzer Handler module has elements to perform several data analyzes and statistics in real time. The Data Handler module is responsible for receiving each of the data lines to construct a FITS image. These 3 modules operate in PC1-PC4, and use shared memory to exchange data. Fingers and Shutter Handlers control these mechanical devices. The Compression Server compresses digital images and creates files with a FITS.ENC extension. Finally, the Control Panel or Observe Program (on PC7), starts and set all the observation parameters. RIDIC also works in this computer and get data from the PC1-PC4 through TCP / IP sockets. This process is detailed below.

B. Client – Server Communication: Sockets

The server software is the one that receives the digital image data; in other words, the server program is RIDIC, which is installed on PC7. The client program is in the Analyzer Data module and is called *scroll.c*. This program sends one by one the lines of digital image data from the Shared Memory. This program was used by an old displaying program called *XDrift* [5], which stopped working due to incompatibility with OS and hardware updates that were made in recent years on PC7.

Communication between client and server is done through sockets that allow the exchange of any data flow reliably and orderly. RIDIC uses flow socket type, and uses TCP transport protocol. This data flow is given by reading line by line, of each CCD. Each data line has the following properties: each pixel has a numeric value that represents the light intensity (counts) of the sky image captured by the detector, this value is assigned to *quint16* (unsigned short int or ushort) data type of 2 bytes (16 bits) size, and whose value ranges between [0; 65,535]. Then, by multiplying 2 bytes / pixel * 2048 pixel = 4096 bytes for each data line that should be transferred through the socket at a predetermined frequency.

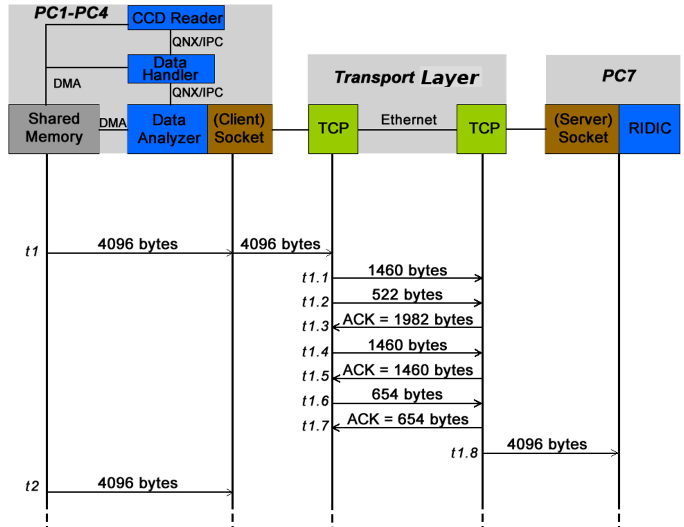

FIG. 6: Temporal diagram of TCP Segments

Figure 6 shows that data line that captures the server program is broken into four segments (in Transport Layer of TCP module) as it exceeds the MTU of 1500 bytes. The sizes of the TCP segments sum 4096 bytes. Each segment received by server program (RIDIC) is stored temporarily in a buffer until total size of the line (4096 bytes) is reached. Then RIDICS displays the line through its GUI.

IV. RIDIC ARCHITECTURE AND OPERATION

A. RIDIC ARCHITECTURE

The component is written in C + + language, using GUI and digital image processing elements of Qt library.

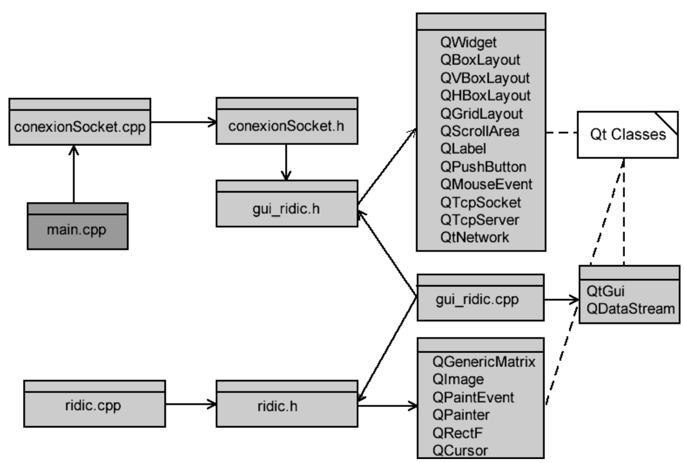

FIG. 7: RIDIC Classes Diagram

Figure 7 showd the RIDIC structure and the dependencies between the different classes that were developed to perform all tasks described in the next section. The software uses basically three classes:

- *"conexionSocket" Class*: It contains functions that establish the connection to the client Socket. After connecting, captured segments are stored in a buffer until it completes

4096 bytes of the data line. *QTcpSocket* and *QTcpServer* classes are used to receive data lines from client program and to send the CCD labels to get data lines from another detector.

- *"ridic" Class:* It contains functions that build and process the digital image from the data lines received by the Socket. *QImage* class is used to construct a digital image representation to access to the pixel data. *QGenericMatrix* class is used to calculate statistics for contrast enhancement. *QPaintEvent, QPainter* and *QRectF* classes are used to paint the data lines.
- *"gui_ridic" Class:* It is responsible for generating the graphical interface. This class uses the "ridic" class to display the digital image. It uses several Qt classes (see Fig. 7) to create the graphic user interface.

B. RIDIC Operation

Tasks performed by RIDIC are seen in the flow diagram shown in Fig. 8. The flow diagram summarizes RIDIC tasks to give a broader perspective and to facilitate their understanding. Although the algorithm may seem simple, within each task, it will perform additional steps that are explained below:

IV.B.1. Read a data line from source:

The source refers to the client program that is installed in PC1-PC4. As we know, the data line (4096 bytes of size) is fragmented by TCP module, and then each segment is stored temporarily in a buffer until the number of bytes received equals 4096. In this case, the buffer is an array of *quint16* (ushort) data type with 2048 length elements. When the vector is full, it is stored in a matrix of *quint16* data type, representing the data array of the digital image, where each cell represents the light intensity of each CCD pixel, and whose range corresponds to a 16 bpp color depth image.

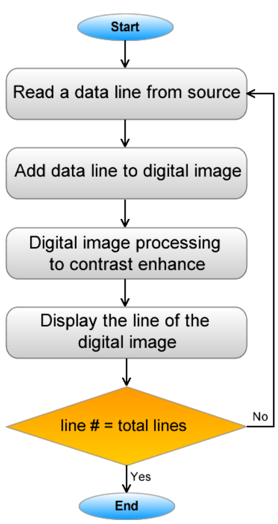

FIG. 8: RIDIC Basic Flow Chart

IV.B.2. Add data line to digital image: To build a digital image using Qt library, *QImage* class is used. This class provides a structure for digital image that allows direct access to pixel data and can be used for displaying by other classes. For the image construction, we preallocated a digital image format. Unfortunately, *QImage* not yet provides a 16 bpp grayscale format (very popular in the astronomical community). Thus, in order to construct the image using *Qimage*, the data had to be re-scaled to a format that supports this class.

A viable option is to re-scale the data from 16 bpp to 8 bpp (2^8 = 256 gray levels), using an 8 bpp direct color format under RGB model. To represent an RGB model in gray scale the following condition must be satisfied (1):

$$R\ Value = G\ Value = B\ Value \qquad (1)$$

Then, we used a 24 bpp color depth format (*QImage::Format_RGB888*), because each component of a 24 bpp image has 8 bits. But, it must comply with condition (1). Thus, the 24 bpp image has not 2^{24} colors, but it has 2^8 gray levels. Each data line read is added to this structure and stored to be processed in the next step.

IV.B.3. Digital image processing to contrast enhancement: The astronomical images have the particularity of being high-contrast, that is, they have very bright objects (stars) in a very dark background (sky background). If an astronomical image is displayed without contrast enhancing (that is, without transforming pixel values), then we obtain an image like Fig. 9 (a), in which only very bright objects are seen in a completely black sky background.

FIG. 9(a): Astronomical image displayed without contrast enhancement.

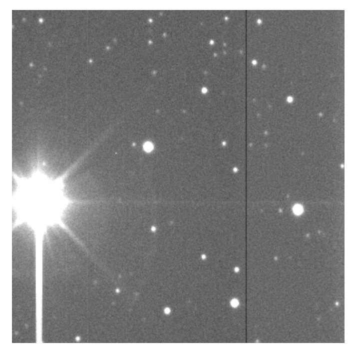

FIG. 9(b): Astronomical image transformed with the Linear Stretch modified method

To solve this problem there are several image processing techniques or algorithms designed to perform contrast enhancement. RIDIC uses a modification of Linear Stretch method [1].

Linear Stretch Modified Method (LSMM): This technique uses digital image histogram, which represents the frequency of the light intensity values (or gray levels) in the image. But, if these values are considered as random variables in [0, 1] interval, then the histogram would represent the probability density function (PDF) [6]. The PDF is a function which describes the probability density at each point in space. The PDF must belong to a probability distribution. We used the sky background as a reference to this study because it always covers the majority of the area in an astronomical image. Sky background is normally distributed [7].

Then, in the original Linear Stretch method, data is rescaled between the minimum and maximum of the image histogram, but in the LSMM, the interval limits will be calculated using the Normal Distribution parameters. It is known that in a normal curve, over 95% of probability is between: $\mu \pm 2\sigma$, where μ is the mean and σ is the standard deviation. Therefore, the new interval limits are $z_1 = \mu - 2\sigma$ y $z_2 = \mu + 2\sigma$, thus obtaining the transformation function shown in equation (2):

$$f(x_{i,j}) = \begin{cases} 0 & \forall x \in [0; z_1) \\ \dfrac{255}{z_2 - z_1}(x - z_1) & \forall x \in [z_1; z_2] \\ 255 & \forall x \in (z_2; 255] \end{cases} \qquad (2)$$

where $x_{i,j}$ represents the pixel at [i, j] position of the digital image. Finally, transforming the astronomical image with this method, we obtain an output image as shown in Figure 9 (b), which presents a significant improvement in the contrast enhancement compared with the unprocessed image. We can see weak bright stars that were undetectable in Fig 9 (a). RIDIC uses this method for astronomical image displaying.

IV.B.4. Display the line of the digital image: For image displaying, we designed a GUI that displays the digital image line by line, that is, it appends a line of 2048 pixels, one below the other, showing only the last 512 lines of drift scan mode observation. The display obtains a "displacement" effect of sky, which can be better appreciated in the following video link: http://www.youtube.com/watch?v=A5WryrM2NAk.

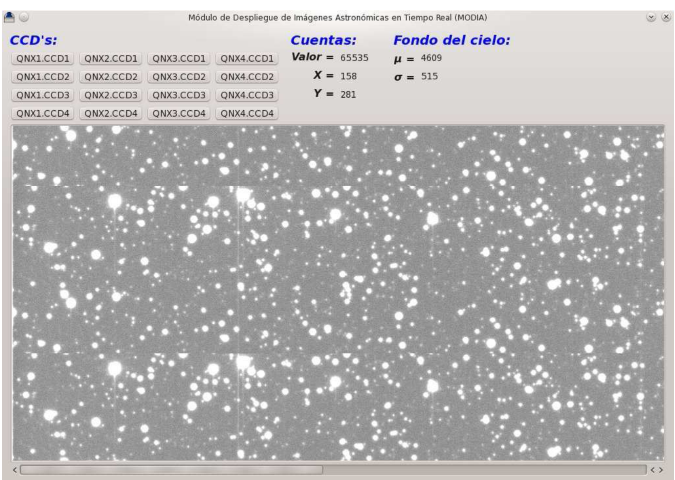

FIG. 10: RIDIC GUI (Spanish interface)

Additionally, GUI has a panel at the top of the window containing three sections (see Fig. 10):

- *CCDs:* This section consists of 16 buttons representing 16 QUEST camera detectors, so that the observer can select the image of any detector, one at a time. Each button sends a message to the program through the Socket server, containing the label of the CCD.
- *Cuentas (Counts):* Provides information about count value (light intensity) of a particular pixel of the image being displayed. This function was achieved using QMouseEvent class offered by the Qt library.
- *Fondo del Cielo (Sky Background):* Provides statistical information of the sky background, such as mean (μ) and standard deviation (σ), which are very useful for user, since it is quantitative information on the sky conditions.

V. Conclusions

RIDIC development is an example of the evolving needs of software needs at CIDA. The importance of achieving technological independence, both in software and hardware, is vital for future work on the telescope Jürgen Stock. At CIDA, there are several projects underway to achieve this goal. The most important of these is the development of electronic control to the new CCD camera, recently acquired by the institution. Another significant project is the migration of the original software that controls the camera, from QNX operating system to Linux. In particular, RIDIC can continue to grow and become a tool of comprehensive analysis of astronomical images quality. To achieve this objective, RIDIC must have algorithms that calculate seeing and the star ellipticity (parameters used in astronomy to assess astronomical image quality) in the astronomical image. Furthermore, in the analysis described in Section III.B we can apply optimization techniques for segmentation at the transport layer, allowing the possibility of displaying more than one image (CCD) at a time, or ideally, to display images of the 16 CCDs of the QUEST camera simultaneously.

References

[1] Jensen, J. R. "**Introduction to Digital Image Processing: A Remote Sensing Perspective**". New Jersey: Prentice Hall. 1996.

[2] Snyder, J. et al. "**QUEST Camera I: A 67 Megapixel CCD Camera Optimized for Driftscan Quasar Discovery**". Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, Vol 3355, pp 635-645.

[3] Downes J. **"Estudio Observacional de la Formación de Objetos Sub-estelares**". Tesis de Doctorado. Universidad Central de Venezuela. Caracas, Venezuela, 2011.

[4] Rodriguez C. "**Estudio y Modificación del Sistema de Adquisición, Almacenamiento y Procesamiento de Imágenes Astronómicas del Telescopio *Jürgen Stock***". Tesis de Maestría en Computación. Universidad de Los Andes, 2007.

[5] Baltay C. et al. "**A Large-Area CCD Camera for the Schmidt Telescope at the Venezuelan National Astronomical Observatory**". The Astronomical Society of the Pacific. Vol. 114, pp. 780–794, 2002.

[6] Gonzalez, R.. "**Digital Image Procesing**". New Jersey: Prentice Hall. 2001.

[7] Vivas, I. **"Modulo de Despliegue de Imágenes Astronómicas para el Telescopio Jurgen Stock del Observatorio Astronómico Nacional de Llano del Hato"**. Tesis de pregrado de Ingeniería de Sistemas. Universidad de Los Andes, Mérida (Venezuela), 2011.