

Distributed Acquisition of Beam-view & Property through Standard Video Components

Sarbajit Pal, Tapas Samanta and Subrata Dasgupta
Variable Energy Cyclotron Centre
1/AF Bidhan Nagar, Calcutta 700 064

Abstract

An inexpensive computerised beam-viewer has been developed exploiting readily available PC add-on h/w and s/w on a Pentium PC running Windows95. A Video Blaster add-on board from popular multimedia systems, is driven to digitise inputs from commercial CCD cameras placed along the beam transport lines. The view acquisition, histogram generation, data compression, intensity computation take place on a 'view_server' on request from 'view_client' across a switched Ethernet LAN. The client collects and displays these on a VB Window, on activation of its control buttons for displaying size, shape, position, histogram, centroid, intensity etc.. Iso-intensity contour generation and thence emittance calculation at the view station is also possible. Service response time is usually around two seconds. The network communications are implemented using SocketWrench / VBX Socket control on Windows 95. Using some freeware s/w's as OLE inside the application, various image processing facilities are incorporated, for enhanced viewing of beam properties. Major portion of the s/w is written in Basic except some compute intensive functions that are written in C.

1 INTRODUCTION

The development is aimed at producing facilities for higher level analysis of the cyclotron beam views, making the data available from any point of the laboratory, exploiting the commercial PC hardware and software products. Subsequently the developments are planned for application in the internal and external diagnostic systems of the newer superconducting cyclotron. The PC add-on Image acquisition boards from video entertainment market have performances acceptable for many usages in our field, when the tv cameras are equipped with some optical processing devices.

2 DEVICES

In the present cyclotron, video cables from 7 beam-viewing stations are already available in the control room. The signal from the station of interest, can be selected through a 8 to 1 video multiplexer unit by controlling 3 output port bits in a digital I/O board.

A PCI-bus video blaster board IE500 from Creative

Labs is used as the video digitizer module. The board accepts NTSC, PAL, SECAM or S-video input and produces 320 x 240 pixels image, giving sufficient spatial resolution for analysis. It handles one-step video capture and storage. The board has an output look-up table for a color palette of 256 colours.

The Windows 95 based Pentium PC in the control room houses this board as shown in Fig.1 and accepts output from the video switcher module. The system developments, though have been done in a way that one device server PC in the vault area will do the image acquisition and analysis for cameras of all the beam-viewing stations. This PC, being connected to the control LAN, will minimize cabling and offer access from other nodes.

Presently, the control room computer screen can locally display the gathered image, upon clicking of a button in its screen. Other computers connected to the LAN can gather and show the analysed beam-view with the help of the developed client server program modules, based on standard network packages.

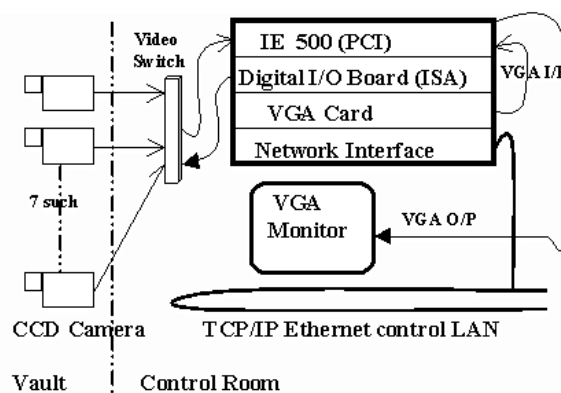


Figure 1. View Acquisition System

3 SOFTWARE

The total software developed comprises of a client and a server part. The server contains

- communication s/w based on TCP/IP and/or UDP/IP protocols using socket functions of the freeware version of the Socketwrench software.
- IEE500 dll's supplied with the card
- image format conversion and image compression s/w developed in-house.

(d) beam-analysis s/w to find the integrated beam current, intensity distribution histograms, beam centroid computation, beam-pipe grid preparations etc..

The client part on the other hand has been written to work in one of the following configurations.

It is composed of the developments as listed down

(a) an user interface screen to select a specific beam viewing station and send a request to the selected station to transmit the current beam-view at selectable intervals.

(b) GUI screen preparation for interactive view of the analysed beam. File manipulations for logging of current beam-view and for selection of stored views.

(c) to accept transmitted composite picture from the server and its decompression.

(d) When needed, further processing of the incoming beam image can be done with the help of GWS.exe, set as an embedded object within the client.

3.1. The Software Architecture.

The camera image is acquired using Asymetrix Digital Video Capture (DVC) Program. This program was embedded in our Capture.exe as an OLE object. This DVC is linked and initialized during the start-up form loading. The properties of DVC like single-frame-capture, 24-bit uncompressed saving of the captured frame and name of the captured image file are set during the time of initialization. All commands sent to DVC including these settings of properties are done by sending keystroke to the DVC application program started as a linked object from the main Capture program. The initialization of the socket control is also done at this stage.

Two sockets have been created on server side. The address family of the both sockets is set as IP address family. Datagram based socket is used as it would not load the LAN to a great extent and binary property is set as image data is to be transmitted. The buffersize property of the socket, is dynamically set for transmission depending upon the size of the file. This is to optimise between data access speed and engaging of LAN bandwidth. After setting these properties, the server initiates a socket-listen action for one socket and wait for an incoming connection on the ports specified by the Local-port. A client application attempting to connect with the server, generates an Accept event. Accepting an incoming connection on that socket, the listening socket connection is closed automatically and the client is connected. This is convenient but limiting the application to single client connection. After serving the client, Action property is set to Socket-listen again. This approach is suitable in this perspective as the request from a client is same in nature, so multiple requests are redundant.

After initialization, the capture program enters a loop and waits for any client's request. On arrival of a request, capture program executes the following functions. It sends series of keystrokes to the embedded DVC program

to collect the image first as 24bit uncompressed BMP format and then converted to 8-bit BMP format image with pallet-map without loss of any information. This compressed bit-map is used for generating axis-wise gray-level histogram used for measuring the beam position and total intensity histogram for beam current calculation. As the header information of the BMP contains the length of image file, other computed properties of the image is appended to compressed beam-profile image for transmission simplicity.

The image data compression and histogram computation is done by an embedded object developed in Visual C++, to enhance the computation. The output file containing image as well as its various beam properties, is broadcast in LAN.

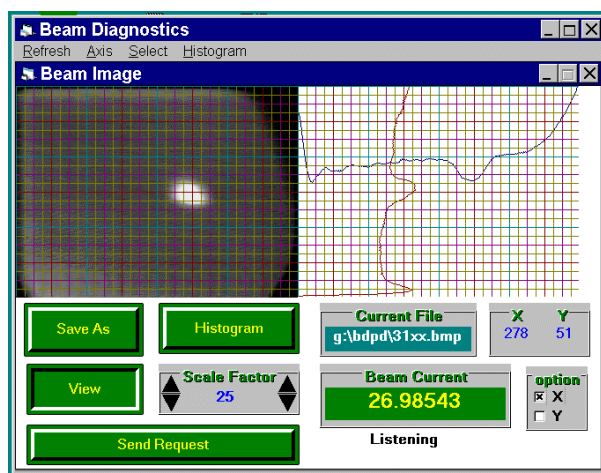


Figure 2. MMI of the server s/w

The server uses Send-length and Send-data property to send captured images with appended properties to the client. The Send-length property specifies the length of the data to be sent to the client. Setting Send-data properties causes the data assigned to it to be written on the socket. As the size of the image data takes several UDP packets, proper care is taken in the code for displaying the image on the client side after receiving all the packets pertaining to a beam image.

The client also has two sockets, one for requesting server for image and another for accepting the incoming datagram coming from the server. The properties are similarly set that of server. These client-server approaches for internal distribution have lossless beam profile data transmission. Here the profile properties can be computed in server as well as client side for off-line analysis. Both server and client posses similar MMI as shown in Fig.2, except that the server has additional capturing ability. Controls like accessing on line or previously stored image, computing beam properties of the image, storing the image with computed properties and on screen visualization of beam position and intensity can be performed from client or server.

Another approach is also implemented which does not

provide much in the way of off-line processing at the client side, but allows simplistic way to access beam-profile in distributed environment. This approach requires installation of PWS (Personal Web Server) at sever PC. The beam image is accessed and displayed, as shown in Fig.3, on the client site by gateway programming. Using Visual InterDev provided by Micro-soft Developer Studio, a Web site is created on the server computer and one HTML page is added to it which contains an IMG tag specifying where the most recently image file is to be found by indicating its path as well as its name with the source. Any client browser accessing that ASP file would be presented with a most recently stored beam-profile image available at the sever which is continuously updated by the sever program after each capture process. As GIF and JPG format is usual graphics file format used in Web pages, the captured BMP formatted images is

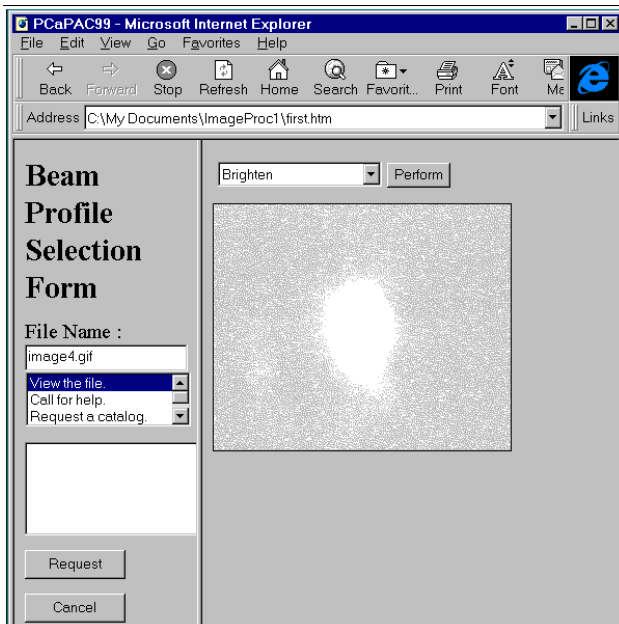


Figure 3. Beam-view through browser

converted to JPG format using Ipvew.exe, a shareware program also embedded in the capture application. The conversion to JPG format here too is done by sending keystroke sequences to an already open program. The main disadvantage of this approach is the lossy compression techniques used for viewing on the browser. But several image processing techniques are implemented for later viewing by Java applet programming. The HTML places two frames on the browser. The first frame named 'Form' holds a user query form helps to select a profile image stored in predefined naming convention. The other frames holds the image loaded interactively on user request. A memory image source is created. This object, used for easy manipulation of raw data, when created, an array contains the raw pixel data is passed to the constructor. This raw data array is manipulated with normal array operations and arraycopy() method of the system is used for faster data copy than a normal 'loop'

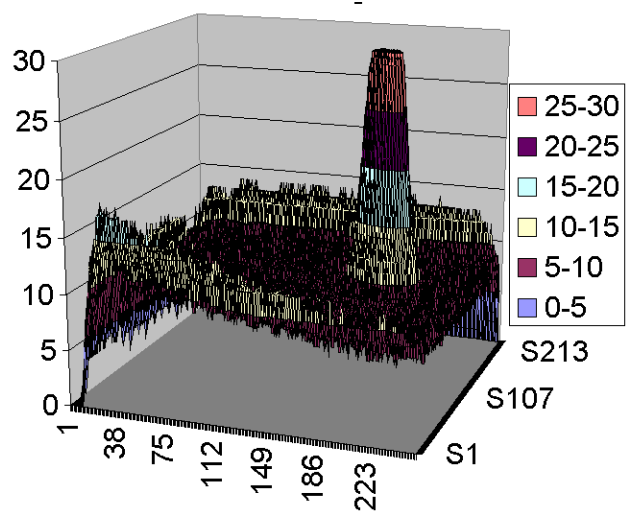


Figure 4. 3D representation of beam

copy. Several image-processing techniques like brighten, dither, segmentation are implemented for enhanced viewing of image on client site.

3.2. Image Analysis

Acquired images are stored in a BMP with 1054 bytes header. After header bytes the image is stored in binary format by skipping the header the binary data is analysed by custom developed code and spread tools (MS Chart) for display and other treatment. The image acquisition system has already given good results in beam analysis. It is installed on the beam lines and acquired 3D representation of a beam stop taken on a diagnostic beam-line is shown in the Fig.4.

4 DISCUSSIONS

The project has been carried out chiefly to ascertain its workability within the environment of our OS and other hardwares and also to complete the software developments necessary for the purpose. After procurement of a digitizer board with higher facilities (eg. better gray level resolution, synchronisation, video multiplexing, antichrominance filter etc.), the current software developments can be directly used to have a higher sensitivity system.

The prototype setup in the laboratory has been successfully tested and we can expect to install it on a real beamline in the first half of 1999. Nevertheless several improvements are required which include:

- realtime acquisition of images, storage in temporary buffer for difference imaging and other image processing.
- Image processing enhancement: presently only beam center statistics finding and total beam current calculation are incorporated.
- Integration of MS chart as a OLE object.
- Micro sample tracking to define region of interest (ROI).

CONCLUDING COMMENTS

Though UDP transmission worked satisfactorily, TCP socket with multiple clients connected with the server is implemented for reliability point of view, sacrificing considerable network bandwidth. While few users are connected for service, this would be better option to offer with high reliability aspect.

REFERENCE

- [1] F. Epaud, P. Verdier, Proc. Int. Conf. On Accel. and Large Experimental Physics Control Systems, Berlin, Germany, October 1993.