International Journal of Technology (2015) 6: 1042-1049
ISSN 2086-9614

DESIGN AND IMPLEMENTATION OF AN AUTOMATIC FACE-IMAGE DATA ACQUISITION SYSTEM USING IP BASED MULTI CAMERA

Wahidin Wahab[1*], M. Ridwan[2], Benyamin Kusumoputro[1]

[1] *Department of Electrical Engineering, Faculty of Engineering, Universitas Indonesia, Kampus Baru UI Depok, Depok 16424, Indonesia*
[2] *Informatics Engineering Study Program, Polytechnic Kampar, Jl. Tengku Muhammad Km.2, Bangkinang, Riau 28461, Indonesia*

(Received: June 2015 / Revised: September 2015 / Accepted: October 2015)

ABSTRACT

Current research trends in 3D Face recognition system requires a special hardware for fast capturing face image data from multi angle view. To support this research, we had designed and implemented an automatic image data acquisition system using multi-camera for capturing facial images from 5° different angle views, which spanned horizontally from 180° from left to right, and vertically from horizontal up to 70° above the face. The system was designed using 30 IP cameras that were mounted on two rigid steel arms that had the form of three quarter of a circle, the two steel arms formed the angle of 90° to each other. At each arm, 15 IP cameras were mounted with 5° spacing vertically to each others. This arm was driven by a DC motor which was controlled by a microcontroller and supervised directly by a laptop computer along with the data acquisition activities. The software for capturing images was designed using C# GUI programming language. The system had been working in good condition and image-data were saved in JPEG format. Time duration of capturing images data for one object face expression with 30 times capturing for the whole angle views, was only 3 minutes 44.5 seconds with total number of 16,650 images collected. The delay time between two cameras capturing was less than 1 sec. This project is aimed to support the 3D face recognition research in the department

Keywords: Data acquisition system; Image data base; Instrumentation; IP camera; Microcontroller

1. INTRODUCTION

The requirements of a good data base containing sets of face images of an observed object or person from various observation angles is crucial in order to have new 3D image processing application software to be tested and verified before it is used in the real implementations. The problems of a long acquisition time to collect such a 3D face images, was that it caused expression changes on the object face due to probably the object or person was feeling fatigue or bored. Too much expression changes of the face would cause inaccuracy and even failure to the algorithm verification. To speed-up the image data acquisition activities, it was required to use multi-camera to capture the images of the object face from different angle of views, and also a strategy to control the data transfer from the multi-cameras to the storage so that all activities occurred within the same time or at least at a very short delay in time.

---

* Corresponding author's email: wahidin.wahab@gmail.com, Tel. +62-21-7270078, Fax. +62-21-7270077
Permalink/DOI: http://dx.doi.org/10.14716/ijtech.v6i6.1848

Many attempts had been proposed to overcome the above problems, Chen et al. (2002) used multi-camera with a special control unit to synchronized the clocking of the capturing time instant of each camera, the method required additional budget and rather expensive. Thiel et al. (2004), suggested method of synchronizations such as, hardware-triggerred and software triggerred using firewire bus, using only 4 (four) cameras. Two AVT Guppy F-036 cameras were connected using IEEE 1394a interface firewire bus and two AVT StingrayF046 were connected using IEEE 1394b interface firewire bus. They had tested the configuration without synchronising the capturing process, and had 10msecs delay in each camera. When they tested the configuration using software synchronisation, the system exhibited even more significant delay reaching 30 msecs, this was due to the waiting time in each of the camera used.

Moore et al. (2010) had shown in their experiments that they could get as small as 5 msec delay between two consecutive cameras capturing, and they had faced with the same delay problem when using the synchronization triggering. The delay problems were due to, firstly the time required for the image capturing and secondly the transmission time required to move the image data to the computer. Duckworth and Robert (2011) suggested to use cameras connected to separate computers for capturing the image, then sent the captured data though internet for reconstructions, collection and processing.

Lina and Benyamin (2003), and Benyamin (2004), used analog movie camera to capture series of face images from 19 view angles in horizontal and 3 view angles in vertical, and had been claimed successfully useful, but the method still required additional effort to select the best image frames recorded in each video recorded and converted them in to the required image file format.

This paper discusses the design and implementation of a fast and automatic facial image capture facility to support the research in 3D face recognition applications. The capturing activities was speed-up using 30 IP based digital Camera, mounted on the two steel frames, the IP cameras are seperated in 5^o view angles so that they covered the view from 0^o to $+75^o$ vertically. The two frames were welded to form a right angle to each other, and moved quickly from view angle 0^o to $+90^o$ horizontally, such that the two frames covered the view from -90^o to $+90^o$ horizontally. A software was written to control the capturing process and the image data transfer to the storage of the main computer so that the delay time between image capture and the total time required were reduced significantly.

2. THE SYSTEM DESIGN

Since the objectives of the project is to reduce the delay time between captured images of each of the camera, then the image capture system was designed using 30 (thirty) IP based digital cameras, mounted on two semi rigid steel semi-circle-shaped frames, which were welded to form 90^o angle and mounted on a vertical steel bar as the axis of rotation, as shown on Figure 1. There were 15 IP Based Cameras mounted on each of the steel frames, separated in 5^o view angles so that they covered the view-angles from 0^o to $+75^o$ vertically.

Each steel frame stood on a rubber wheel at the end of its leg, to make it moved quickly within a semi-circle, and one of these rubber wheels was driven by a DC motor under closed loop speed-controlled by a microprocessor to be synchronized to the capturing event and supervised by the main computer or laptop. The stopping positions were indicated by black dots on the floor, so that the control system became simplified and therefore it required only a simple proportional control with constant gain only.

2.1. The System Architecture

The system architecture is shown in Figure 2, using a computer or a laptop as the main computer to supervise and synchronize every activity in the application. The IP based cameras were connected through the hub-switch using independent IP address for each camera. A DC motor with gearbox attached was used to rotate the rubber wheel to move the steel-frame to the pre-marked position quickly for the face-image capturing. The DC motor was powered by a motor driver and controlled by a microprocessor-based controller, the target position was read by the reflective opto-sensor, which was fedback to the microcontroller when the position was reached. The start from the beginning to the end of the image capture process was supervised completely by the main computer.

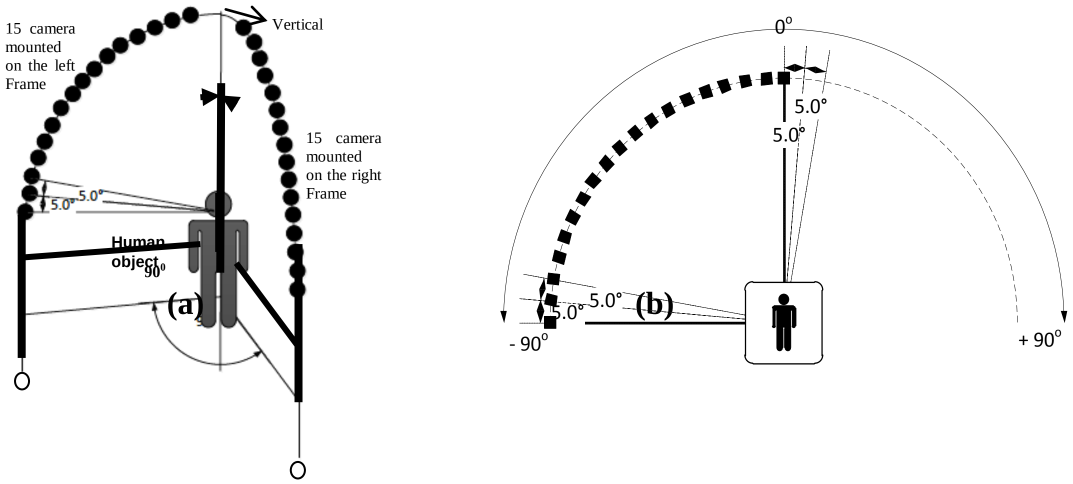

Figure 1 (a) Perspective view of the steel frame for camera mounting; (b)Top view showing the black-dots on the floor separated 5^{o} each

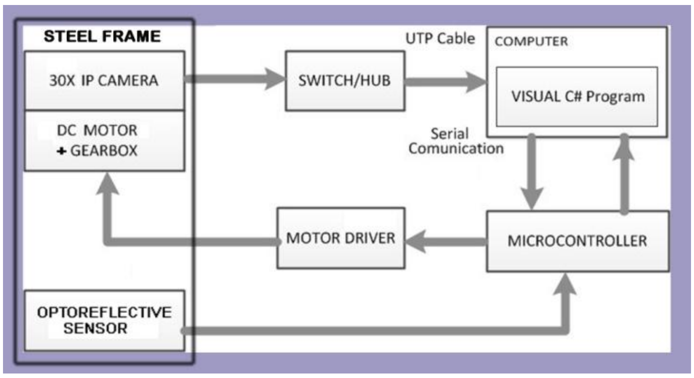

Figure 2 The system architecture

The main computer first initialized all the hardware, and set up IP connection to all 30 cameras, then it initiated a "start capture signal" to start all cameras to capture images at once, after all cameras finish capturing data, it initiated a signal to the motor driver to start the motor to quickly bring the steel-frames to the next 5^{o} target position. It also initiated the transfer of the image data to the ram memory of the main computer.

After the microcontroller received a feedback signal from the opto-reflective sensor indicating that the required position had been reached, then it stopped the motor and sent a "position reached" signal to the main computer, and the main computer directly initiated a start capture signal to the IP cameras again to start capturing images. This sequence was repeated until all required view angle had been reached, then the computer initiated a signal to the micro controller to start the motor driver to bring the steel frames back to its home position and meanwhile it saved all the captured image-data on to the hard disc drive in the computer.

2.2. The IP based Camera

The IP based Camera is essentially a digital CCTV Camera with an embedded microprocessor within it, such that it was already made with the required capturing software inside it, and therefore it can be easily configurable for, such as, a specific IP address, the resolution mode, username and password, image format etc. The IP based camera had provided the standard TCP/IP protocol, such that it could be access using the web-browser though the internet using the standard http protocol or rstp protocol. The IP based camera is capable of recording image in JPEG format or recording video in M-JPEG format, it can be set to have 640×480, 320×240 and 160×120 resolutions, to suit the required applications.

The IP based camera was used in this project due to the following reasons, it has a reasonably fast speed of data transfer, they can be connected in parallel with almost unlimited numbers of cameras which can be started independently or all together at the same time, the cost is economically lower in comparison to most other similar class of camera, the IP address for each camera can be set individually such that each camera can be access independently. The output can be in jpeg format for image data or in mjpg format for video streaming data. In this project we recorded the face-images using the mjpeg format. Since it is buffered inside the camera memory, and therefore prohibited the lost of image data captured during the acquisition process.

2.3. The DC Motor Controller

The steel-frame arm where all the cameras are attached was moved from one desired view angle to another by using a DC motor as the actuator. And since the objective is the speed of motion to the desired position, then the DC motor was configured to be under simple closed-loop speed control configuration, and using the PWM method to set the reference speed such that it could be easily controlled to reach the desired position using the microcontroller. The pulse width modulated (PWM) pulses used, had a constant frequency but the duration of each pulse was in proportion to the analog voltage output of the motor driver module, and as such the speed of the dc motor would be in proportion to the width of the pulse. Using an 8-bits control code to the pwm motor driver, it can provide 256 variation of code, then it can provide the value from 0 to 255 representing for 0% to 100% duty cycle of the pwm output pulses. By carefully repeating the speed set-up experiments, we could set the right code to the microcontroller and hence the best required motor speed could be set constant for the whole image capturing process. The Position control is shown in Figure 3. The analysis of its mathematical model was fairly straight forward and had been given in detail in the research report (Thiel, 2004) and therefore was not discussed in here.

A low cost micro controller AVR8535 was used as the controller and pre-programmed using the Codevision-Avr, the target position was marked using black dots on the floor as seen in Figure 1(b), indicating every 5^o target position. The reflective opto sensor was used to read the black-dot positions, the response from the reflective opto sensor was read-in from port#A and the control command to the motor was sent through port#C, while port#D was used to connect to the main computer using USB type connector for data transfer to the Driver Motor module where the pwm circuit was built.

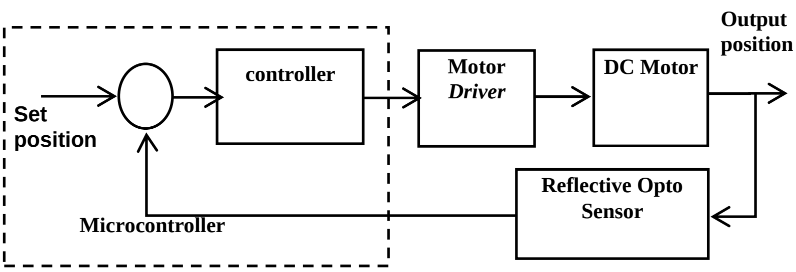

Figure 3 The position control system

3. THE SUPERVISORY SOFTWARE

The completed system was supervised and monitored using a set of software, written in visual Studio version 2010, this software was loaded into the main computer to control all the image capturing process. In this application we used only a personal computer or a laptop with i7 processor, which controlled all 32 cameras to start or stop operating, the start and stop of the frame position controller and to manage the saving of the captured image-data into the storage of the computer. The flowchart of the software in both the main computer and the microcontroller was shown in Figure 4, where it clearly showed the synchronization between the two processors.

The supervisory software was designed using its graphics user interface (GUI) to easily provide the monitoring of the captured images on the computer screen, and to supervise the capturing process of the 32 IP based cameras. As soon as the software was activated, it directly initialised the hardware, the microcontroller of the frame position controller, and all the IP-cameras addresses. It set the camera to record using the Mjpeg format. The IP Network address of the cameras are set as the following,

Http:/192.168.0.N/, where N =1, 2, 3, ... or 32

And the recorded mjpeg buffered data can be read from the camera buffered memory using IP address as,

URL : http://192.168.0.N/video.cgi, where N =1, 2, 3, ... or 32

Where the mjpeg format was recorded as a streaming video using the collection of a number of jpeg images, the total number of jpeg images captured was set to 30 (thirty) frames of images, this value can be set by user request. The captured data was read through the IP network and then stored in to the hard disc in the computer, using a specific filename defined as follows,

cam_N_stop_K_20141016_161751_499_cap_X

1 2 3 4 5 6

where field #1 is the number of the camera, and N=1, 2, 3 up to 32, field#2 is the stop position, K=1, 2, ...19, field #3 is the recording date, field #4 is the numeric value of record time (hour, minute and second), field #5 is the milli seconds of the record time, and field#6 is the number of capture sequences, where X=1, 2, up to 30.

By having the time of recording up to milli seconds, we could calculate the length of time required for the recording and reading the image-data from each of the camera, and hence calculate the total length of time required for one cycle of image data acquisition for all view angles and positions (Ridwan, 2014).

When the next black-dot position was reached, the microcontroller sent back a signal to indicate this condition, and then the supervisory software re-started the next capturing process at this view angle. The activities was repeated for each view angle by moving the steel frame to each of the black-dot position, when all positions had already been done then the supervisory software initiated the motor to move the steel frame back to its home position and be ready to do the image capturing process for the next object. The capturing process was done in the following order as shown in Table1.

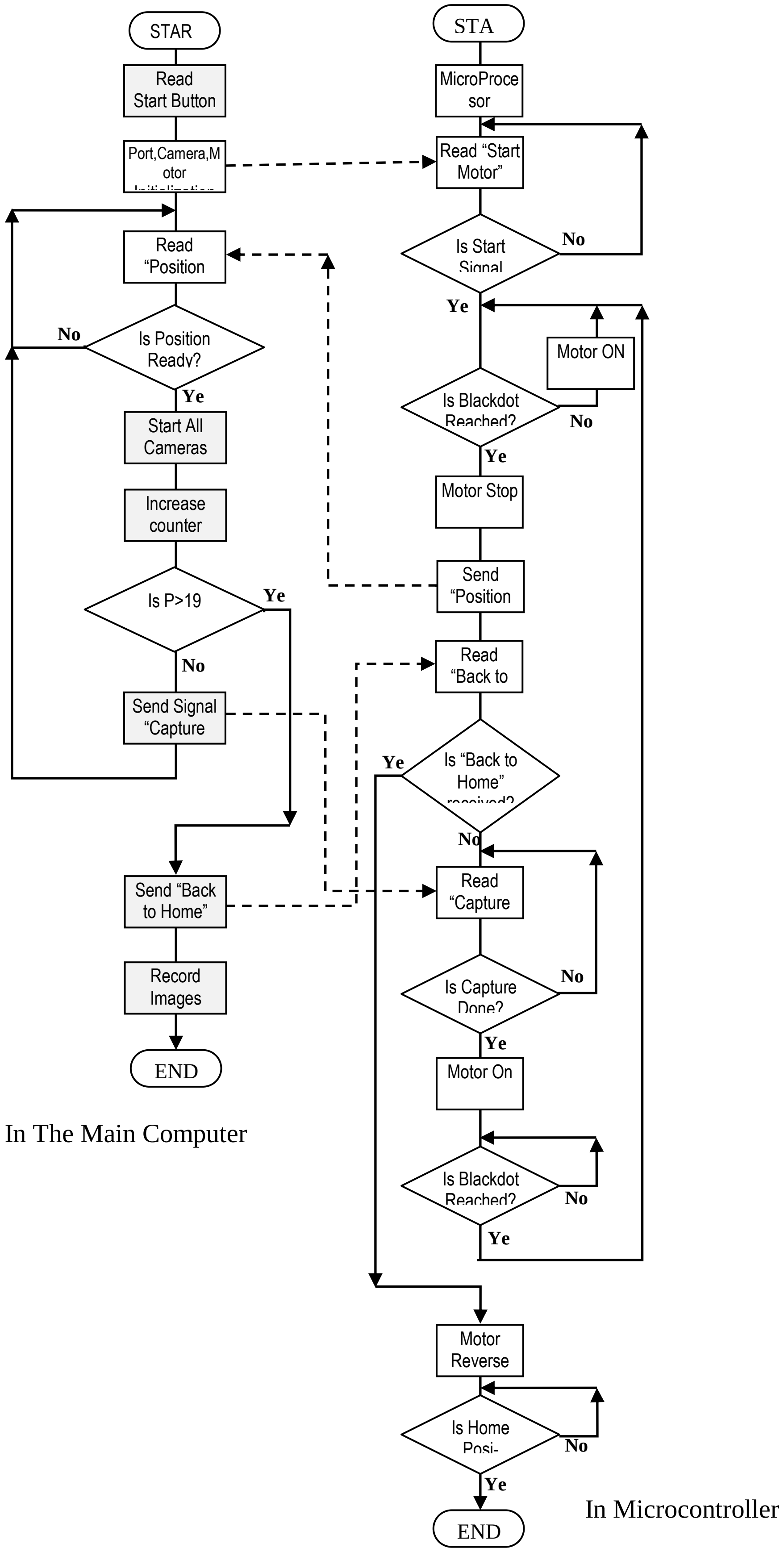

Figure 4 The supervisory software flowchart

Table 1 The sequence of image capture activities at each view angle and positions

Black-dot position	View Angle (degrees)	Right-arm of Steel Frame Activated cameras	View Angle (degrees)	Left-arm of Steel Frame Activated cameras
#1	-90^o	Camera #16 to #30	0^o	Camera #1 to #15
#2	-85^o	Camera #16 to #30	5^o	Camera #1 to #15
#3	-80^o	Camera #16 to #30	10^o	Camera #1 to #15
...	...	...	...	...
...	...	...	...	...
#17	-10^o	Camera #16 to #30	80^o	Camera #1 to #15
#18	-5^o	Camera #16 to #30	85^o	Camera #1 to #15
#19	0^o	No recording	90^o	Camera #1 to #15

From the experiments of the acquisition process it was found that the total time required was 3 min 44 secs and 292 msecs, using a laptop with i7 processor and 30 cameras with 160×120 resolutions, recording 30 frames/camera in mjpeg format with the total number of 16,650 images recorded. On average the time for each image captured is less than 1 secs and the time to move the frame between two consecutive positions was 1.4 secs. Parts of the image-recorded is shown in Figure 5 with the indicated positions and view angles. The images can also be viewed using other image editor software such as Adobe Photoshop, Art-view, Microsoft picture viewer, etc.

Figure 5 Captured Images (partly) from selected view angle and positions

4. CONCLUSION

The project was successfully built at the department of Electrical Engineering University of Indonesia to support the 3D face recognition system research at the department. The image capturing activities was done in a relatively short time, for one complete cycle of capturing it took only about three and half minutes. Within this short time it was expected that there shouldn't be any noticeable emotional expression changes on the object being captured. This is crucial in having the reliable data-base to support the research.

The project had successfully utitized the buffering inside the IP camera and the available time the movement of the steel frames from one position to the next one, so that the wasted time had been used efficiently for the data transfer from the camera to the main computer. The speed of movement of the steel frames are also limited by the structure of the frames, higher speed would generate vibrations in the steel frames as it moved, and when it happened then it needed time to wait until the vibration died out, before the capturing process could start, or else it would destroy the images quality. The average time between each image capture was less than 1 sec.

It was seen that the images were not limited around the face only but still covered a larger area than required, and also the position of the head of the image was somewhat different in position inside the image. It is suggested to provide the zoom facility to the camera to adjust the center of area and the focus area of the captured images.

5. REFERENCES

Benyamin, K, 2004. Pengembangan Hemisphere Structure of Hidden Layer Neural Networks dan Optimasi Strukturnya Menggunakan Algoritma Genetika. *Prosiding Semiloka Teknologi Simulasi dan Komputasi serta Aplikasi*, Fakultas Ilmu Komputer, Universitas Indonesia [in Bahasa]

Chen, T., Kung, S., Lin, Y 2002. *Frame Synchronization in a Multi-camera System*. New York: AT&T Corporation

Duckworth, T., Robert, D.J., 2011. Camera Image Synchronisation in Multiple Camera Real-time 3D Reconstruction of Moving Humans. *IEEE/ACM 15th International Symposium on Distributed Simulation and Real Time Applications (DS-RT)*, pp. 138–144

Lina, Benyamin, K., 2003.Sistem Pengenalan Wajah 3-D Menggunakan Penambahan Garis Ciri Pada Metode Perhitungan Jarak Terpendek dalam Ruang Eigen. *MAKARA*, *SAINS*, Volume 7(1), pp. 1–10 [in Bahasa]

Moore, C., Duckwortht, T., Aspin, R., Robert, D., 2010. Synchronization of Images from Multiple Cameras to Reconstruct a Moving Human. *Distributed Simulation and Real Time Applications (DS-RT), IEEE/ACM 14th International Symposium*, pp. 53–60

Ridwan, M., 2014. *Sistem Akuisisi Data Citra Otomatis dengan Multi Kamera Berbasis IP*. Master Thesis, Dept. E.E., Universitas Indonesia [in Bahasa]

Thiel, D., Goulart, M., Botelho, S., Bem, R., 2004. *Hardware and Software Infrastructure to Image Acquisition using Multiple Camera*s. Universidade Federal do Rio Grande – FURG