Embedded-Vision Solution to Interpret and Integrate Point-of-Care Clinical Diagnostics

Blessy Navara[1]
M.tech - Embedded Systems
Raghu Engineering College
Visakhapatnam, India
blessynavara91@hotmail.com

Mr. Sivaramarao Bodda[2]
Associate Professor- Electrical Engineering
Raghu Engineering College
Visakhapatnam, India
sivaram.eee@raghuenggcollege.in

***Abstract*— Embedded vision is the ability of an embedded system to take logical or intelligent decisions based on the images captured and processed by the system. Point-of-care diagnostics provide clinical diagnostic facilities to the patient at the site, thereby reducing time taken for diagnosis and enhancing the quality of treatment. The proposed system enables medical professional to interpret these point-of-care clinical diagnostics and connect the different remote diagnostic sites, so that they can be accessed, monitored, controlled and maintained by a trained medical professional.**

Keywords— RDTs, Embedded vision, RaspberryPI, OpenCV, GTK+, LAMP

I. Introduction

Ubiquity of digital communications is helping us to communicate with people in the remotest parts of the earth. However, people in remote regions are affected with some of the most infectious diseases like malaria, dengue, typhoid, and more and are not able to access reliable clinical diagnosis. Early diagnosis of such diseases can help prevent further spread of infection. According to WHO statistics nearly, 214 million malaria cases and an estimated 438,000 malaria deaths are recorded in year 2014-2015 alone [1]. These statistics show that diagnosis of such infectious diseases can help prevent deaths, by encouraging the affected people to take in the right medication. Traditional laboratory methods implemented for clinical diagnosis requires complex, high-end laboratory equipment, skilled technicians and experts, experienced medical professional. An automated approach for reading microscopic slides can be seen in [11]. It would take ages, for all these resources to reach the remote regions. So, the lack of skilled labor and equipment is taking toll on the people who are deprived of primary health care. That’s where point-of-care diagnostics came to the rescue. Point-of-care clinical diagnostics offer diagnostic facilities to patients at the site. Diagnosis for such infectious diseases requires no high-end equipment. Point-of-care diagnostics are reliable most of the time. The global point of care (PoC) diagnostics market is anticipated to reach USD 20.9 billion by 2024 [9]. One such point-of-care diagnostic solution is Rapid Diagnostic Test, coined as RDT. RDTs are Lateral Flow Assays which give a visual interpretation of the test in place.

Few drops of blood and buffer would trigger a reaction to take place on RDT. A strip of region on RDT is coated with the corresponding antigen. If the test in place is malaria, then the strip of region on RDT is coated with say pGluDH. As soon as blood comes in contact with this coated region on RDT, there would be a change in color indication. If the parasite is present in the blood, they would react with this chemical to give a bright red color. Absence of parasite in the blood is indicated by no change in the color of chemical coated region. Cell-phone Based Devices (CBDs) are developed to meet this purpose as in [2] and [3]. A smartphone dongle developed for this is seen in [10].

But why is embedded vision needed for point-of-care diagnostics, if point-of-care clinical diagnostics are already solving the issue. There are few major problems associated with rapid diagnostic tests used in point-of-care diagnostics.

- Width of chemical coated strip on RDT is very small, yet is very much visible to the human eye.
- Red band formed after chemical reaction is not always dark enough to be visible to the human eye.
- Decision is highly subjective, which proves it very dangerous in case of false diagnosis.

So, embedded vision here plays an important role in interpreting the rapid diagnostic tests. Work done by [4] shows that the result can also be quantified. Fionet developed a device as seen in [5] and GE globalresearch is developing a device [6], [7], [8] to read its own RDTs.

But next, why do we need to integrate these point-of-care diagnostics?

Point-of-care diagnostics is spread across remote regions. Elementary and primary health care centers provide primary screening for the above mentioned infectious diseases. There is a need to connect these centers and their corresponding diagnostics. Skilled and experienced medical professional cannot be always present in these primary health centers. But, supervision is necessary in any case. In general, there is hardly any staff in primary health centers, and even if present, they are not skilled or trained to handle situations which requires one to take immediate critical decisions. If point-of-care diagnostics are supervised under a trained medical professional either directly or indirectly, it prevents any false diagnosis, prevents error. Further, by connecting these point-of-care diagnostics, one can maintain Electronic Health Record (EHR) of all the tests being taken. By maintaining the electronic records, a doctor can remote login from a different geographical position and access these results.

So, using embedded vision, we can interpret Rapid Diagnostic Tests used in point-of-care diagnostics. Thereby, we minimize

978-1-5090-4620-1/16/$31.00 ©2016 IEEE

human errors. This project also aims to integrate different such devices, so that they can be monitored, controlled and maintained by a skilled, trained personnel. The proposed embedded system is designed to take advantage of ubiquity of digital data and communication, to reach the remote regions and provide primary health-care. The proposed embedded system is designed to be portable, to consume less power.

II. IMPLEMENTATION DETAILS

The paper is organized into the following subsections.

1. System design
2. GUI design and integration into the embedded system
3. Customized embedded vision algorithm using OpenCV to read and interpret RDTs.
4. Storing the patient details and result to the server

A. System Design

The proposed system contains the following: an embedded processing device, imaging device, display device, mouse and keyboard for input.

Fig. 1: Hardware Setup showing Raspberry PI2B

Proposed system described here is implemented on a Raspberry Pi 2B embedded board. It comes with a Broadcom BCM2836 SoC which hosts 900 MHz 32-bit quad-core ARM Cortex-A7 processor with ARMv7 architecture. A 15-pin MIPI camera interface (CSI) on-board connector is used to connect with the Raspberry Pi camera. The board has 4 2.0 USB ports. Video out can be observed from the HDMI. A micro SD card can be inserted into the Micro SDHC slot for onboard storage. Board can be connected to the Internet through 10/100Mbps Ethernet USB adapter.

The reason for choosing Raspberry pi is

- The board is based on Linux operating system (which makes development easier)
- It supports Ethernet connectivity, on-board USB, CSI (Camera Serial Interface to connect the camera)
- It is compact – well suitable for an embedded system
- It draws minimum power even when running with camera and input devices connected to it
- It is easily affordable.

Camera and input devices draw current from the board. The SoC(Silicon-on-chip) used in this board supports the vision libraries necessary for image processing. Since the operating system is a Linux distribution, we used GTK+ libraries required for development of GUI. RDT holder is a simple mechanical arrangement to place the RDT in right orientation with the camera.

Fig. 2: Complete experimental setup showing RDT holder

B. GUI Design and integration into the embedded system

GTK stands for GNU Image Manipulation Program Tool Kit (GTK). A user-friendly GUI is required to take in the patient details and guide the user through the testing process. GTK is chosen for this device, because the device is based on embedded Linux. GTK+-2.0 libraries provide different GUI elements. GUI is designed to contain widgets for labels, text entry, and to display captured image and results obtained. Widgets are aligned using vertical and horizontal boxes. However, GUI is designed to be simple, so that even a layman can be guided through the process and effective to avoid any faults.

A “NEW TEST” widget would begin the process of taking in the details and capture image to interpret the rapid diagnostic test. Details captured from the user are; Patient’s name, age, ID, Gender and type of test to read. After taking all the details, camera is automatically triggered. User need not push any button to activate the camera. Shell scripts are integrated into the program so that the process becomes automated. The device hardly takes any time to obtain the result after running the customized embedded-vision algorithm. GUI displays the processed result whether the result is positive or negative or invalid, by analyzing the captured image. Captured image is also displayed on the GUI. Results and other patient details are simultaneously uploaded to the server through another shell script which runs immediately after image processing.

Fig. 3: Patient details fill-in widget of GUI implemented with GTK+

Fig. 4: Test result and image displayed in GUI

C. Customised Embedded vision algorithm to read and interpret RDT

Images are captured using 5MP Raspberry Pi camera. Original image captured from camera contain high-frequency noise. A 3x3 Gaussian filter is used to remove high frequency noise by smoothening out the original image. Each channel is histogram equalized so as to equalize changes in light. The 3-channel RGB image is converted into a single channel gray scale image for further processing.
The subject of interest in captured image is the red band formed due to the reaction between chemical coated strip and the parasite, if present. Intensity of this red band is not constant for every sample under test. Intensity depends on concentration of parasite in the given sample. Results for color based image segmentation were not always accurate for the above stated reason. So color based segmentation could not be used in this algorithm. Since the system should be able to operate in any working conditions, light is a primary concern. However, minimum illumination is required.

So a modified threshold algorithm called adaptive threshold is used in the proposed system. In this algorithm, threshold value varies for each individual pixel based on the gray level values of surrounding pixels. Width of red band is generally 0.5mm to 1mm. So a 5x5 matrix of surrounding pixels is considered for every pixel for which a threshold is calculated. Threshold value for each pixel is calculated using the formula

$$T_{(x,y)} = \frac{1}{n^2} \sum_{i=x-2}^{x+2} \sum_{j=y-2}^{y+2} P_{(x,y)} - c \qquad (1)$$

Where,
$P_{(x,y)}$ is the pixel value at (x, y).
N is the value to define a nxn square matrix around the pixel at (x, y).
C is a constant subtracted from the mean weighted average.
$T_{(x,y)}$ is the threshold value calculated for the pixel at (x, y).

Threshold operation is done using the formula

$$if\ P_{(x,y)} > T_{(x,y)}\ then\ P_{new\ (x,y)} = Max\ value \qquad (2)$$

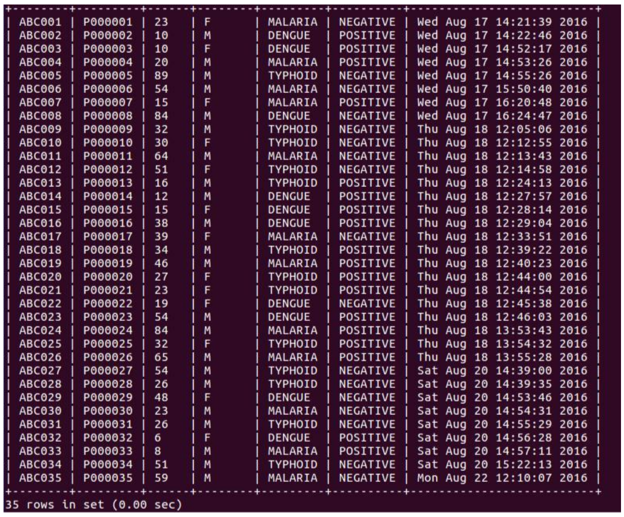
$$else\ if\ P_{(x,y)} > T_{(x,y)}\ then\ P_{new\ (x,y)} = 0$$

Where, Max value is 255

Each pixel is changed to 0 or 1 based on the threshold value calculated using eq. (1).
After the above operation, a binary image is obtained. Now, the subject of interest is in white pixels on a background of black. To extract the band, connected components are calculated on the obtained image. Connected components are sequences of linked pixels. Here, sequences are extracted by finding the pixel values linked to each other. By calculating the number of sequences returned, the system takes a decision if the result is positive or negative or invalid. Results are time-stamped and then sent to the server

D. Storing patient details and result to the server

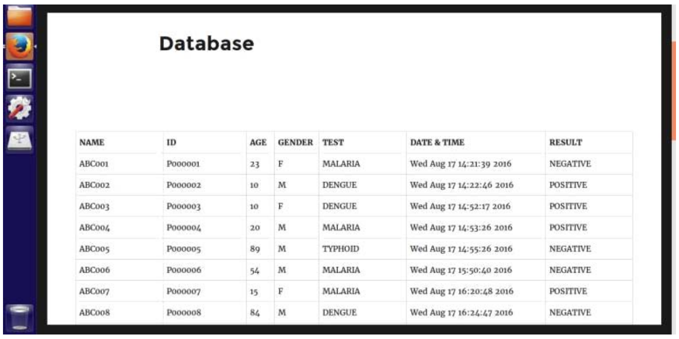
A LAMP server is designed on Ubuntu 14.04 32-bit system. Apache is used as server, Mysql is used as database and PHP is used for scripting.



ABC001	P000001	23	F	MALARIA	NEGATIVE	Wed Aug 17 14:21:39 2016
ABC002	P000002	10	M	DENGUE	POSITIVE	Wed Aug 17 14:22:46 2016
ABC003	P000003	10	F	DENGUE	POSITIVE	Wed Aug 17 14:52:17 2016
ABC004	P000004	20	M	MALARIA	POSITIVE	Wed Aug 17 14:53:26 2016
ABC005	P000005	89	M	TYPHOID	NEGATIVE	Wed Aug 17 14:55:26 2016
ABC006	P000006	54	M	MALARIA	NEGATIVE	Wed Aug 17 15:50:40 2016
ABC007	P000007	15	F	MALARIA	POSITIVE	Wed Aug 17 16:20:48 2016
ABC008	P000008	84	M	DENGUE	NEGATIVE	Wed Aug 17 16:24:47 2016
ABC009	P000009	32	M	TYPHOID	NEGATIVE	Thu Aug 18 12:05:06 2016
ABC010	P000010	30	F	TYPHOID	NEGATIVE	Thu Aug 18 12:12:55 2016
ABC011	P000011	64	M	MALARIA	NEGATIVE	Thu Aug 18 12:13:43 2016
ABC012	P000012	51	F	TYPHOID	NEGATIVE	Thu Aug 18 12:14:58 2016
ABC013	P000013	16	M	TYPHOID	POSITIVE	Thu Aug 18 12:24:13 2016
ABC014	P000014	12	M	DENGUE	POSITIVE	Thu Aug 18 12:27:57 2016
ABC015	P000015	15	F	DENGUE	POSITIVE	Thu Aug 18 12:28:14 2016
ABC016	P000016	38	M	DENGUE	POSITIVE	Thu Aug 18 12:29:04 2016
ABC017	P000017	39	F	MALARIA	NEGATIVE	Thu Aug 18 12:33:51 2016
ABC018	P000018	34	M	TYPHOID	POSITIVE	Thu Aug 18 12:39:22 2016
ABC019	P000019	46	M	MALARIA	POSITIVE	Thu Aug 18 12:40:23 2016
ABC020	P000020	27	F	TYPHOID	POSITIVE	Thu Aug 18 12:44:00 2016
ABC021	P000021	23	F	TYPHOID	POSITIVE	Thu Aug 18 12:44:54 2016
ABC022	P000022	19	F	DENGUE	NEGATIVE	Thu Aug 18 12:45:38 2016
ABC023	P000023	54	M	DENGUE	POSITIVE	Thu Aug 18 12:46:03 2016
ABC024	P000024	84	M	MALARIA	POSITIVE	Thu Aug 18 13:53:43 2016
ABC025	P000025	32	F	TYPHOID	POSITIVE	Thu Aug 18 13:54:32 2016
ABC026	P000026	65	M	MALARIA	POSITIVE	Thu Aug 18 13:55:28 2016
ABC027	P000027	54	M	TYPHOID	NEGATIVE	Sat Aug 20 14:39:00 2016
ABC028	P000028	26	M	TYPHOID	NEGATIVE	Sat Aug 20 14:39:35 2016
ABC029	P000029	48	F	DENGUE	NEGATIVE	Sat Aug 20 14:53:46 2016
ABC030	P000030	23	M	MALARIA	NEGATIVE	Sat Aug 20 14:54:31 2016
ABC031	P000031	26	M	TYPHOID	NEGATIVE	Sat Aug 20 14:55:29 2016
ABC032	P000032	6	F	DENGUE	POSITIVE	Sat Aug 20 14:56:28 2016
ABC033	P000033	8	M	MALARIA	POSITIVE	Sat Aug 20 14:57:11 2016
ABC034	P000034	51	M	TYPHOID	NEGATIVE	Sat Aug 20 15:22:13 2016
ABC035	P000035	59	M	MALARIA	NEGATIVE	Mon Aug 22 12:10:07 2016

35 rows in set (0.00 sec)

Fig. 5-a: Screenshots of database showing the patient details and test result



Database

NAME	ID	AGE	GENDER	TEST	DATE & TIME	RESULT
ABC001	P000001	23	F	MALARIA	Wed Aug 17 14:21:39 2016	NEGATIVE
ABC002	P000002	10	M	DENGUE	Wed Aug 17 14:22:46 2016	POSITIVE
ABC003	P000003	10	F	DENGUE	Wed Aug 17 14:52:17 2016	POSITIVE
ABC004	P000004	20	M	MALARIA	Wed Aug 17 14:53:26 2016	POSITIVE
ABC005	P000005	89	M	TYPHOID	Wed Aug 17 14:55:26 2016	NEGATIVE
ABC006	P000006	54	M	MALARIA	Wed Aug 17 15:50:40 2016	NEGATIVE
ABC007	P000007	15	F	MALARIA	Wed Aug 17 16:20:48 2016	POSITIVE
ABC008	P000008	84	M	DENGUE	Wed Aug 17 16:24:47 2016	NEGATIVE

Fig. 5-b: screenshot of webpage accessed by a medical professional

As soon as results are obtained, a program sends the patient data and corresponding image to the server. Server database is updated every time a new test result is obtained. Samples which are already tested in laboratory are checked on this device and the results are uploaded to database. Doctor login from a different region would show the above database.

III. Experimental Analysis

We configured the Raspberry pi board with the required version of operating system, and libraries, i.e. OpenCV, GTK+. OpenCV provides the necessary libraries for image processing. Camera is integrated with the board in such a way that it is triggered only after all patient details are filled in. All the different modules of software and hardware are tested recursively to obtain a fault-proof system.

Fig. 6: Image after processing showing the identified red band which is displayed in green color

Fig. 7: Binary image showing the region of red band

After image processing, high frequency noise is removed to obtain a smoothened image. Modified adaptive threshold algorithm used in this system gives the accurate result. As seen in fig.6 the red band is identified and, marked in green color. We tested each sample recursively to determine system's reliability and repeatability. Binary image is shown in fig.7 for a negative sample of Malaria test.

IV. Algorithm

The proposed automated system implements the following steps. Steps 1-3 are to be done by the user.

1. Placing RDT in the RDT holder
2. Clicking "NEW TEST" widget on GUI to take in the patient details
3. Filling in the Patient details
4. Camera is triggered automatically after patient details are filled in.
5. Image is captured and saved to the device.
6. Preprocessing: removal of high frequency noise using Gaussian smoothening filter.
7. Histogram equalization of each channel to make up for the variations in intensities of light.
8. Conversion of 3-channel image to single channel gray scale image
9. Applying modified adaptive threshold to obtain a binary image, thereby isolating the subject of interest.
10. Finding the linked pixels in the binary image to extract the sequence, i.e. the required red band.
11. Decision making: based on the sequences extracted from the binary image, system takes the decision about the test in place.

V. Conclusion

Above proposed system can be used to read, interpret and maintain Electronic Health Record (EHR) at a low cost in point-of-care diagnostics. Relying on embedded vision to interpret RDTs makes the system fault-proof and reliable. By integrating the system to server, it helps in maintaining Electronic Health Record (EHR). A medical professional can remote-login from a different geographical position to monitor, trace and can even keep a quality check. Thus, the device can aid in point-of-care diagnosis.

References

[1]. http://www.who.int/features/factfiles/malaria/en/

[2]. *Sandeep Kumar Vashista, Onur Mudanyalib,E.Marion Schneidere, Roland Zengerlea, and Aydogan Ozcanb* Cellphone-based devices for bioanalytical sciences *Anal Bioanal Chem*. 2014 May ; 406(14)

[3]. *Sandeep Kumar Vashist, E. Marion Schneider and John H.T. Luong* Commercial Smartphone-Based Devices and Smart Applications for Personalized Healthcare Monitoring and Management in *Diagnostics* 2014, *4*.

[4]. *Donald C. Cooper, Bryan Callahan, Phil Callahan & Lee Burnett* Mobile Image Ratiometry: A New Method for Instantaneous Analysis of Rapid Test Strips in Nature Precedings · January 2012

[5]. http://fio.com/our-solution/ "Fionet embeds data capture into routine care delivery"

[6]. http://www.geglobalresearch.com/blog/researchers-collaborate-to-improve-malaria-diagnostic-test

[7]. http://www.geglobalresearch.com/blog/accurately-analyzing-malaria-tests-difficult-conditions-matter-life-death

[8]. http://www.geglobalresearch.com/blog/collaborative-approach-will-improve-malaria-detection-test

[9]. http://www.grandviewresearch.com/industry-analysis/point-of-care-poc-diagnostics-industry

[10]. *Laksanasopin T, Guo TW, Nayak S, Sridhara AA, Xie S, Olowookere OO, Cadinu P, Meng F, Chee NH, Kim J, Chin CD, Munyazesa E, Mugwaneza P, Rai A, Mugisha V, Castro AR, Steinmiller D, Linder V, Justman JE, Nsanzimana S, Sia SK* "A smartphone dongle for diagnosis of infectious diseases at the point *of care". Science Translational Medicine 04 Feb 2015:Vol. 7, Issue 273, pp.*

[11]. S. S. Savkare, S. P. Narote, "Automated system for *malaria parasite identification" published in Communication, Information & Computing* Technology (ICCICT), 2015 International Conference.