Computer Vision based Detection System

D. Gowri Shankar
Electronics and Communication
SRM University, Kattankulathur (T.N)

R. Manohari
Dept of Electronics and Communication
SRM University, Kattankulathur (T.N)

Abstract

The Internet and wireless broadband infrastructure is adding an extra edge to the next generation video surveillance. Besides enhancing the ability to improve security, it will also help improving productivity, customer satisfaction and regulatory compliance of the business. A noninvasive video surveillance system has been developed successfully using the ARM 9 single board computer as the development platform. The hardware comprises of the Friendly ARM mini2440 SBC, customized IR sensitive camera, Wi-Fi Module. The software implementation is based on the Linux kernel and Qt framework with porting of cross-compiled OpenCV and GUI libraries. Owing to the use of open source technologies and choosing embedded Linux as the development platform, the development cost has reduced tremendously. The embedded system target platform used in this paper is Samsung S3C2440 which based on ARM9 embedded processor core. The Linux of released version is not fit the hardware of embedded system, so the cross-develop environment is needed to customize Linux operating system. It describes the methods and progress of transplanting the embedded Linux to the target board based on the S3C2440 processor, including the establishment of cross-compiler environment, the reduction and compilation of start-up code (bootloader) and Linux kernel 2.6 and the construction of root file system with the point focused on the structure and function of bootloader as well as the transplantation is feasible and using OpenCV (Computer Vision) library, the motion detection application is developed.

Index Terms

Motion detection, ARM, OpenCV, Linux, Threshold match algorithm, contours.

1. INTRODUCTION

As the video monitoring system developing more than 20 years, it experiences from analog to digital. The hardware

and software are correspondingly improved in the functional and performance, but many deficiencies still exist. The original control is manual operation. Artificial monitoring system can consume quite a lot of manpower and material resources and not be remotely controlled, and the function is also single, which will result to that

D.Gowri Shankar is a postgraduate student at SRM University, Kattankulathur, India (e-mail: gshankar.rdg@gmail.com).

Manohari.R is with the Department of *Electronics and Communication Engineering*, SRM University, Kattankulathur, India (e-mail: *manohari.r*@ktr.*srmuniv*.ac.in).

the error report rate is high, it can only alarm on the spot and cannot achieve real-time remote alarm to reduce losses and the working situation of record. These drawbacks make alarm system will fail to alarm and alarm error information, alarm system security and utility will seriously reduce. The video monitoring system needs to monitor and detect the work locations, according to the acquired image information, so it is very important to process the image. This proposal will focus on the introduction of image acquisition and moving image detection according to the collected images [2]. Computer vision has found widespread acceptance in mobile applications like video streaming, smart cameras, vehicle navigation, smart traffic light systems, and virtual reality. The increasing demand for video applications like context aware computing on the mobile embedded systems like cell phones requires the use of computer vision algorithms. The implementation of computer vision algorithms is computationally intensive and resource exhaustive. The system engineer has a mandate to optimize these algorithms so as to meet real time deadlines. Fortunately engineers can take advantage of sophisticated features available on mobile processors like dual core architectures. This can be achieved by selecting the proper mobile processor to improve performance of the algorithms by using different on-chip resources. When dealing with specialized mobile processors like Samsung S3C2440, Open CV algorithms can be modified to take advantage of the available resources to speed up computation. Open CV is an open source computer vision library developed by Intel Corporation. It is highly optimized to take advantage of Intel architecture enhancements like Advanced Vector Extensions, Streaming SIMD Extensions and MMX technology [1].

2. OVER ALL SYSTEM DESIGN

The project features a web camera, installed in house premises, which is operated by an advanced information and communication system based on Internet, Java programming, Scheduler modeling and mobile phone mentoring. The web camera sensors are connected within house and software module is designed to be devoted to analyze the signals which are generated by it. The camera detects motion of any intruder in front of the camera dimensions. The software module communicates to the intended user either via Internet network, or via mobile services. This system is designed to be a powerful information and security tool for residence as well as office premises. Inputs are generated when our hardware subsystem becomes aware of a threat via the connected sensors (USB Camera) and alerts the software sub-system that some motion is detected, when an intruder is found at house premises. The hardware subsystem (ARM9), which is constantly monitoring with the help of connected sensors, then provides real time inputs to software subsystem which then processes these inputs to trigger different events.

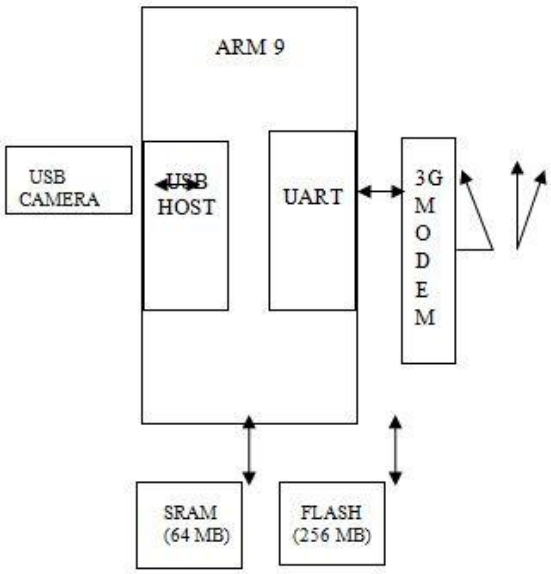


Fig. 1: Overall System design

3. SYSTEM SOFTWARE DESIGN

The software structure of the system, it mainly includes transplanting the operation system of ARM-Linux, compiling camera driving program, calling display, storage and compression programs.

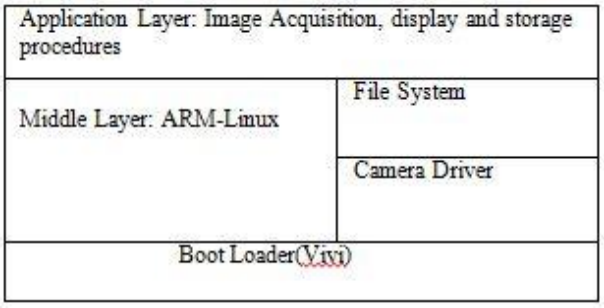


Fig 2: Software development

4. System overview

A. Embedded Platform

The MINI2440 Development Board is based on the Samsung S3C2440 microprocessor. Its PCB is 4-layer boarded, equipped with professional equal length wiring which ensures signal integrity. All MINI2440 boards are manufactured in mass production and released with strict quality control. On startup it directly boots preinstalled Linux by default. The MINI2440 development board is a 100 x 100(mm) board equipped with a wide variety of connectors, interfaces and ports.

B. OpenCV

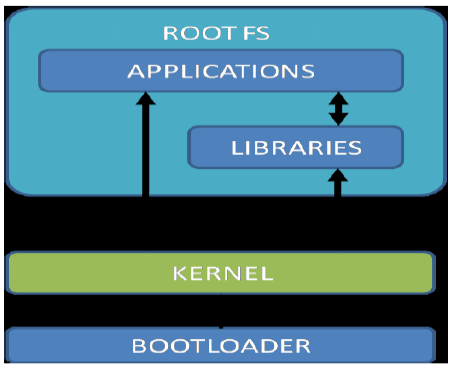
OpenCV is an open source computer vision library. The library is written in C and C++ and runs under Linux, Windows and Mac OS X. There is active development on interfaces for Python, Ruby, Matlab, and other languages. OpenCV was designed for computational efficiency and with a strong focus on real time applications. OpenCV is written in optimized C and can take advantage of multi core processors. If you desire further automatic optimization on Intel architectures [Intel], you can buy Intel's Integrated Performance Primitives (IPP) libraries [IPP], which consist of low-level optimized routines in many different algorithmicareas. OpenCV automatically uses the appropriate IPP library at runtime if that library is installed. One of OpenCV's goals is to provide a simple-to-use computer vision infrastructure that helps people build fairly sophisticated vision applications quickly. The OpenCV library contains over 500 functions that span many areas in vision, including factory product inspection, medical imaging, security, user interface, camera calibration, stereo vision, and robotics. Because computer vision and machine learning often go hand-in hand , OpenCV also contains a full, general-purpose Machine Learning Library (MLL).This sub library is focused on statistical pattern recognition and clustering. The MLL is highly useful for the vision tasks that are at the core of OpenCV's mission, but it is general enough to be used for any machine learning problem[3].

C. Linux Platform

A minimum Embedded Linux system is composed of three essential components:

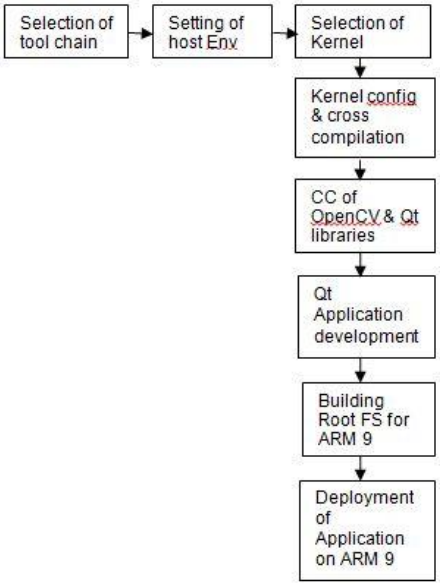
- Bootloader: It gets the operating system loaded and running on the board.
- Kernel: It is the software that manages the hardware and the processes.
- Root File System: It is a directory tree containing needed libraries, scripts, utilities and applications. It follows a standard convention of file system hierarchy, for example, /usr,/bin,/etc etc.
- Application: The program that runs on the board. The application can be a single file or a collection of hundreds of executables.

After the boot loader loads the kernel and the kernel mounts the root file system, it looks for a program to execute by default, or the user can supply the program with given environment variables. This program runs as the first process and it continues to run. When this process stops, the kernel, and thus the entire system, stops running. On the desktop Linux system, this program is likely init, which is known as the first user process. The init process performs low level initialization and system modes. In embedded devices various startup scripts are written to initialize the user program as the environment variables are not specified in the init script.All these components are interrelated and thus depend on each other to create a running system. Working on an embedded Linux system requires interaction with all of these, even if our focus is only on the application.



5. Algorithm

The system is initialized to grab an image. According to the system's inbuilt timer, the next image is grabbed which is referred to as Current.



If there exists a difference in the Old and Current images' RGB values, the motion is said to occur which is detected by the Sensors. Here motion detection algorithm is based on frame difference calculation in terms of RGB values and brightness threshold values stored in byte arrays. The algorithm compares two consecutive frames Old Current, pixel by pixel to generate a difference value. If the difference value is greater than a fixed value (randomly taken), then motion is detected. Else if, there is no difference between previous and current frame's byte arrays then Old is set to Current. The process repeats according to the program's set timer.

Three frame difference algorithms are used for image processing. The method observe the threshold size of moving obhects and moving objects under the condition and then judge the difference image threshold to identify whether there is a moving object.

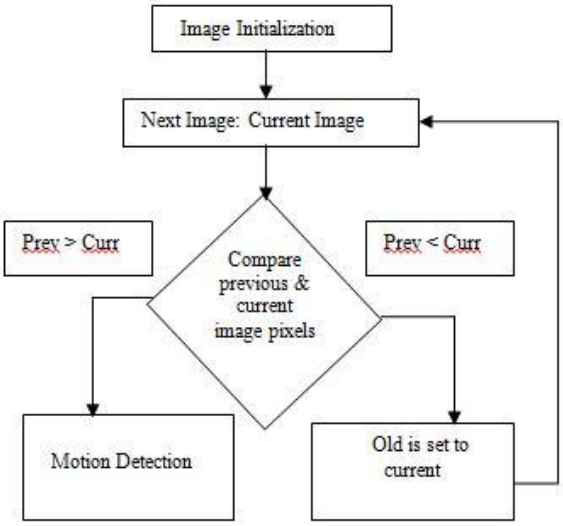


Fig. 5 Threshold match Algorithm

Results

Arm-Linux-gcc and Linux kernel is configured and compiled for the target board. Installation and compilation of OpenCV is completed. Using OpenCV, video is monitored and output image is obtained.

Two sets of video monitoring will be obtained.

- Output Image: Video will be monitored, still image will be displayed.
- Difference Image: Here, contour image will be displayed.

Contours are represented in OpenCV by sequences in which every entry in the sequence encodes information about the location of the next point on the curve.

Obtained images are displayed below:

1. When there is no motion, no detection has done.

Fig. 6a

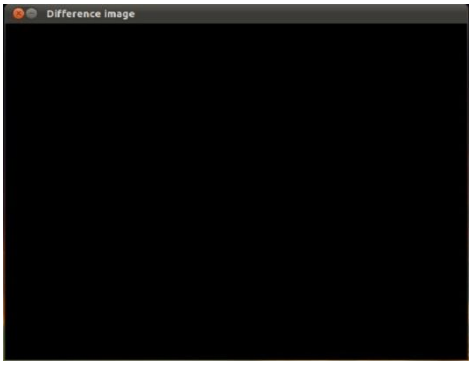


Fig. 6b

1. When there is motion, contours are displayed.

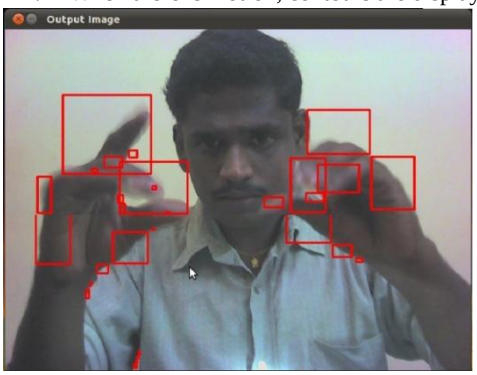


Fig. 7a

Fig. 7b

Conclusion

Upon arriving at a conclusion the software sub-system runs its pre-programmed checklist of appropriate responses. This includes posting the intruder's snapshot to the mobile phone via MMS.

Thus in our project work we have obtain the introduction of image acquisition and moving image detection according to the collected images. These collected images are developed on the ARM9, OpenCV and implemented in Linux platform. By collecting the images we can reduce the losses and working situation of record. This thesis of combining ARM and Open CV with Linux platform can be implemented in any video surveillance system. Such systems can replace the systems which are used in areas like ATM centers, Shopping mall, Banks, Educational Institutes, Government Offices etc.,

REFERENCES

[1] P.K, Anumol Jose, Bibin Jose, Dinu L.D, Jomon John, Sabarinath G, "Implementation and Optimization of Embedded Face Detection System", *International Conference on Signal Processing, Communication, Computing and Networking Technologies* (ICSCCN 2011).

[2] Zhang Xiaozhi,"Image processing and detection based on ARM+Linux platform",*International Journal of Image Processing, Volume (2) Issue(l),2012.*

[3] G. Bradski and A. Kaehler, Learning OpenCV, OReilly Publications,2008.

[4] Li Haifeng. "*Based on streaming video motion information analysis system research and implementation [D], Jilin University, 2009".*

[5] *Open Source Computer Image Library Reference Mannual.2001 (8).*

[6] *Victor Gedris," An Introduction to the Linux Command Shell or Beginners" In Co-Operation With: The Ottawa Canada Linux Users Group and Exit Certified.*