

Real Time Smile Detection using Haar Classifiers on SoC

Srikanth Sriram

M.E. Student, Department of Electronics and
Communication Engineering
Vardhaman college of engineering, Telangana,
India

Babu Illuri

Assistant professor, Department of Electronics and
Communication Engineering
Vardhaman college of engineering, Telangana,
India

ABSTRACT

Smile detection in real time video is an interesting problem with many potential applications. This paper is intended to implement a real time smile detection from video using Haar Classifiers through Raspberry Pi BCM2835 processor which is a combination of SoC with GPU based Architecture. For capturing video we used Raspberry Pi Camera Board of 5MP and which plugs directly into the CSI connector on the Raspberry Pi. Computer vision OpenCV libraries with python IDE is used for face detection and smile detection through linux based raspbian operating system. Frame rates of various video resolutions during smile detection in Raspberry Pi are also observed. The present method can be used in low cost robotic computer vision applications, human computer interaction (HCI) and in education as well because cheap and small hardware used.

Keywords

Smile Detection, SoC, Raspberry Pi, Computer Vision, Haar classifiers, OpenCV.

1. INTRODUCTION

Human facial expressions have the ability to communicate emotion and regulate interpersonal behavior and facial expression of emotion was a natural, flexible, and physiological response which could provide evidence of an individual's internal mental state. A smile is the most common facial expression that occurs in people's daily life. It often indicates pleasure, happiness, appreciation, or satisfaction. Detecting smiles can be used to estimate a person's mental state. Smile detection has many applications in practice, such as interactive systems (e.g., gaming), product rating, distance education systems, video teleconferencing, user interfaces and patient health monitoring. For example, the statistics on the spectators smile can be a hint for "how much the spectators enjoys" the multimedia content.

This paper focuses on real time smile detection captured on real world scenarios[1]. It presents a real time approach to smile detection, in which haar like features applied to the gray scale images are used as simple features. Adaboost algorithm then is taken to choose and combine weak classifiers based on haar like features to form strong classifier for smile detection. In the experiments the data set that is trained by the real life scenario images with OpenCV and Python IDE is used for face detection and smile detection based on haar classifiers. System-on-chip (SOC) approach for the detection of smile using raspberry pi and raspberry pi camera module is implemented. A 5 mega pixel camera module is used for image acquisition and Raspberry Pi system is used to process the given data and show it to the display. We also observed the frame rates of real time video when smile detection takes place with various resolutions with Raspberry Pi computer.

2. TO CONSIDER RETROSPETIVELY

The problem related to smile detection is facial expression recognition. There are many academic researches on facial expression recognition. Recently Whitehill et al. [2] presented a comprehensive study on practical smile detection. They collected the GENKI database consisting of 63 000 real-life face images from the Web. He used different type of image representations used for their accuracy of real time situations in smile detection.

Caifeng Shan et al [3] presented smile detection by boosting pixel differences in which the intensity differences between pixels in the grayscale face images are used as features. Smile detection has received much interest for commercial applications. For example, in some digital and mobile cameras, the "smile shutter" shoots automatically when a smiling face is detected. Modern digital camera have the smile mode function and camera takes picture automatically when a smiley face is detected and also have the ability to make a perfect group photo by shooting continuous frames and then select the best smiling face of each person and make a perfect group photo.

But building a real time smile detection prototype with low cost and low power small size hardware is tough and challenging problem for many researchers and students for their facial expression based robots and commercial applications. Yen chang [4] presented the low cost real time face detection, tracking and recognition for human robot interactions. In this he used BeagleBoard C4 module as main hardware it cost about 125\$ And designed social health care robot.

Table 1. Various SoC based development boards with their prices.

S.No	SoC based development boards	Price
1	Raspberry pi	35\$
2	BeagleBone Black	45\$
3	BeagleBoard	125\$
4	UDOO	100\$
5	Intel MinnowBoard Max	99\$

Now there is large SOC based development boards are designed to provide relatively Low-cost development platform for hobbyists to try out the powerful new system-on-a-chip (SOC) devices that were essentially capable of performing all the duties of a computer on a single chip. So compared to the other development boards with raspberry pi .It is low cost and affordable for anyone who can use it for projects or computer. The comparison of various development boards price are Shown in table 1.

3. SMILE DETECTION USING HAAR CLASSIFIERS

Object Detection using Haar feature-based cascade classifiers is an effective object detection method proposed by Paul Viola and Michael Jones [5]. Their paper is “Rapid Object Detection using a Boosted Cascade of Simple Features”. They used an Adaboost [6] machine learning based approach here a cascade function is trained from a lot of positive interested and negative non interested images. It is then used to detect objects in other images. Like faces detection, cars detection, stop sign detector[7] etc. For determining whether or not the person is smiling facial expression, we used the same principle we used to detect smile in human faces.

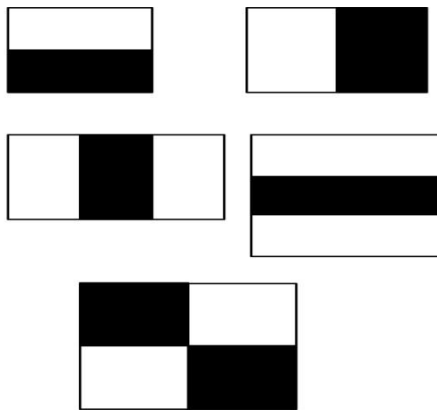


Fig 1: Haar like features

Here we will work with smile detection. Initially, the algorithm needs a lot of positive images (images of smile faces) and negative images (images without smile faces) to train the classifier. Then there is need to extract features from it. For this, haar features shown in figure 1 image are used. They are just like our convolutional resource management. Each feature is a single value obtained by subtracting sum of pixels under white rectangle from sum of pixels under black rectangle.

Now all possible sizes and locations of each kernel is used to calculate plenty of features. (Even a 24x24 window results over 160000 features). For each and every feature calculation, we need to find sum of pixels under white and black rectangles. To solve this, we use the integral images. It minimizes the calculation of sum of pixels, how large may be the number of pixels, to an operation involving only four pixels. It makes things super-fast. But among all these features we calculated, most of them are irrelevant. To find the most relevant features in given image we use Adaboost algorithm for the selection of use full features.

For this, we apply each and every haar feature on all the training images. For each and every feature, it finds the best threshold which will classify the smile faces[8] to But

obviously, there will be errors or misclassifications. We select the features with minimum error rate, which means they are the features that best classifies the smile and non-smile images. In this process each image is given an equal weight in the beginning. After each classification, weights of misclassified images are increased. Then again same process is done. New error rates are calculated and also new weights. The process is continued until required accuracy or error rate is achieved or required the number of features are found. Several Haar feature classifiers compose a stage. A stage comparator sums all the Haar feature classifier results in a stage and compares this summation with a stage threshold. The threshold is also a constant obtained from the AdaBoost algorithm. Each stage does not have a set number of Haar features. Depending on the parameters of the training data individual stages can have a varying number of Haar features. Final classifier is a weighted sum of these weak classifiers. It is called weak because it alone can't classify the image, but together with others forms a strong classifier. The entire process is shown in figure 2.

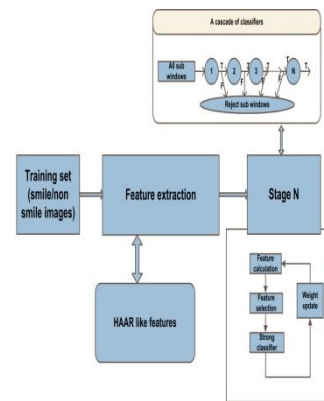


Fig 2: Haar Classifier Training Process for Smile Detection

For detection of our own smile detector we used the Identical haarcascade training parameters of width=43,height=19, number of stages=16, stage hit rate=0.995, stage false alarm rate=0.5, weak classifier decision stump tree depth=1, weight trimming rate=0.95. Our data set is based on version 0.4 of the Semi-supervised haar training of a fast&frugal open source zygomatic smile detector [9]. hear they taken smile and non smile images from real life image conditions. They taken images from LFW manual and GENKI database. This data set which used above consist of 16 stages and 621 features in total with area under the curve performance accuracy of 90.21% smile detector. This is use full for real time applications with real life conditions.

4. REAL TIME SMILE DETECTION MECHANISM USING OPENCV

For smile detection mechanism we used the rapid object detection algorithm proposed by Viola and Jones is used as the basis of our design. These specific haar features of a smile mouth region used for smile detection. Figure 3 shown above shows the real time smile detection mechanism used in our prototype.

In the process of real time smile detection mechanism the given image or video taken frame by frame and converted into gray scale. Varying illumination is one of the difficulties for smile detection in real-life faces. We used histogram equalization to the gray scale image because HE is a simple

and widely used technique for normalizing illumination effects.

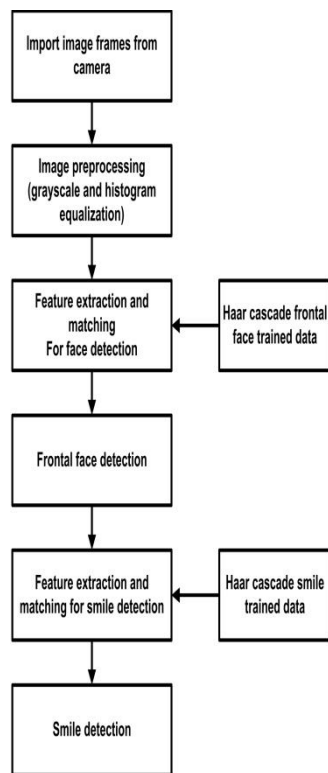


Fig 3: Real time smile detection system mechanism

System is capable of detecting the smile from the captured image for the purpose of prototype. From the above Section, the system first look for face detection then smile detection process takes place. Face detection determines where in an image a face is located and it is done by scanning the different image scales and extracting the exact patterns to detect the face. A face candidate is a rectangular section of the original image called a sub-window. Generally these sub-windows have a fixed size (typically 24×24 pixels). The Prototype is to built with Haar-Like Feature function from OpenCV [10]. Haar classifier detection is used to create a search window that slide through a image and check whether a certain region of an image looks likes face or not. Haar like features and a large set of very weak classifier uses a single feature to define a certain image as face or non face. The data is stored at haar cascade frontal face data. Each feature is described by the template its coordinate relative to the search window origin and the size of the feature. The search window quickly scanning the first classifier on the cascade as shown in the Figure 4, if the classifier returns false then the computation on that window also ends and results no detected face(false). Moreover, if the classifier returns true, then the window will be passed down to the next classifier in the cascade to do the exact same thing. When all classifier return true for that window, then the result will returns true also for that certain window face detected.

After the face detection process smile detection takes place. Smile detection determines where in a face image a smile is located and it is done by scanning the different face image scales and extracting the exact patterns to detect smile.

For the smile detection we used haar –like features and function from OpenCV. In this haar classifier detection is

used to create a search window(43*19 pixels) that slide through a face image and check weather a certain region of a face looks for person smiled or not. A smile posed roi candidate is a rectangular section of the original image called a sub-window.

Haar like features and a large set of very weak classifier uses a single feature to define a certain face image as smile or non smile which are stored at haar cascade smiled data. Each feature is described by the template its coordinate relative to the search window origin and the size of the feature. The search window quickly scans the first classifier on the cascade of classifiers. if the classifier returns false then the computation on that window also ends and results no detected smile(false).Moreover, if the classifier returns true, then the window will be passed down to the next classifier in the cascade to do the exact same thing. When all the classifier return true for that window, then the result will returns true also for that certain window smile detected. Haar classifier algorithm scans the image and creates a bounding box as returns for each detected smile in given face.

5. SOC BASED SMILE DETECTION SYSTEM DESIGN



Fig 4: Broadcom. BCM2835 SoC (System on Chip) Based raspberry pi board

Raspberry Pi (represented in Figure 4) is a credit-card-sized single-board computer developed in the UK by Raspberry Pi [11] foundation with the intention of stimulating the teaching of basic computer science in schools. It has two models; Model A has 256MB RAM, one USB port and no network connection. Model B has 512MB RAM, 2 USB ports and an Ethernet port. It has a Broadcom BCM2835 SOC(system on a chip) which includes an ARM1176JZF-S 700 MHz processor, Video Core IV GPU, and an SD card. The GPU is capable of Blu-ray quality playback, using H.264 at 40Mbits/s. It has a fast 3D core accessed using the supplied OpenGL ES2.0 and OpenVG libraries. The chip specifically provides HDMI and there is no VGA support. The foundation provides Debian and Arch Linux distributions and also Python as the main programming language.

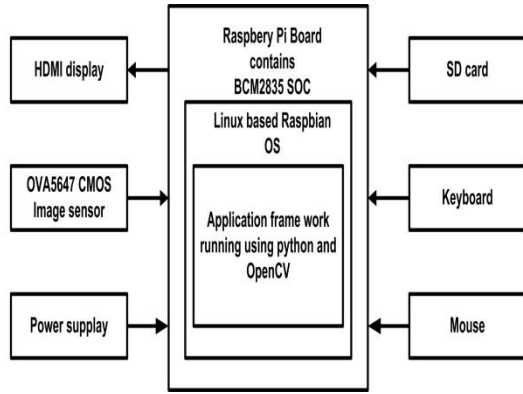


Fig 5: Smile detection prototype block diagram

The smile detection mechanism as described above was implemented on a Raspberry Pi BCM2835 SOC computer. The Raspberry Pi was the hardware of choice because of its low power requirements (3.5W), its small size, and its very low cost (under US\$35). The smile detection prototype block diagram shown in figure 5.

5.1 Hardware Setup

We used 8GB sd card for Raspberry Pi and installed the Raspbian operating system through Win32Diskimager through windows computer. Mouse and keyboard used and placed into the two USB ports of Raspberry Pi [12]. Micro USB based Samsung mobile phone charger(5.0 V, 0.7A) used for power the Raspberry Pi. We used the raspberry pi camera module (OVA5647 CMOS Image Sensor) for image acquisition. the camera module is capable of up taking photos up to 5 megapixels(5MP)(2592*1944 pixels) and can record a video at resolution up to 1080p30(1920*1080*30fps). The camera module is an Omni vision 5647. It is comparable to cameras used in mobile phones.

The camera board is a small PCB that connects to the CSI-2 camera port on the Raspberry Pi using a short ribbon cable [13]. It provides connectivity for a camera capturing still images or video recordings. We installed drivers of camera board from the internet using Ethernet cable of Raspberry Pi. Next we connect the HDMI display to Raspberry Pi by the HDMI port. Experimental Hardware setup is shown in figure 6.



Fig 6: Hardware setup

5.2 Software Setup

We installed the Raspbian OS in the SD card of Raspberry Pi. Raspbian is a free operating system based on Debian optimized for the Raspberry Pi hardware. We used python as main programming language for Raspberry Pi. In order to develop the software for smile detection it is required to use integrated development environment (IDE) in the raspberry Pi. To perform the image processing task with raspberry pi we installed and used OpenCV library developed by intel. Which is compatible with Raspberry Pi. To run the Raspberry Pi camera board with OpenCV. We installed Video4Linux drivers. And we programmed in python with OpenCV using haar like feature function from OpenCV. Then implemented the real the time smile detection mechanism on Raspberry Pi.

In this smile detection process camera board used for real world image acquisition then the raspberry pi used for process the given information through python programming and OpenCV libraries and signal to the real world by HDMI display. The figure 5 shows block diagram our smile detection prototype.

6. RESULTS

The result of smile detection is shown in figure 7 and figure 8. First we used the still face of smile and non smile photos in the Raspberry Pi to detect the smile. Then we used the camera board with raspberry pi to detect the smile of person in a real time video. The results were shown by HDMI display connected to the Raspberry Pi. In the real time video if the person appears first the face is detected that is represented by drawing green rectangle at the frontal face. Then if the person smiles smile is detected. That is represented by drawing a blue rectangle at mouth region. We also calculated the frames per seconds that Raspberry Pi is taken for smile detection. We performed operation with different video resolutions in real time. That is shown at table 2 performing low resolutions videos we achieved good frame rate but we achieved less smile detection like in 240*160 resolutions in video. At High resolutions video frame rate is low compared to low resolution video but accuracy improved.

The results are compared with the graphical form of various video resolutions with low frame rates when smile detection takes place is shown at figure 9. By that the BCM2835 SOC can be more useful for real time facial expression computer vision applications with low cost preference. For high resolutions we can achieve accurate smile detection but frame rate is low.

Table 2. Experimental results of various video resolutions with frame rates when smile detection takes place.

Resolution of the video	Frames per seconds(fps)
800*600	0.4 - 0.8
640*480	0.4 - 0.8
320*240	1.4 - 4.0
240*160	4.0 - 8.0

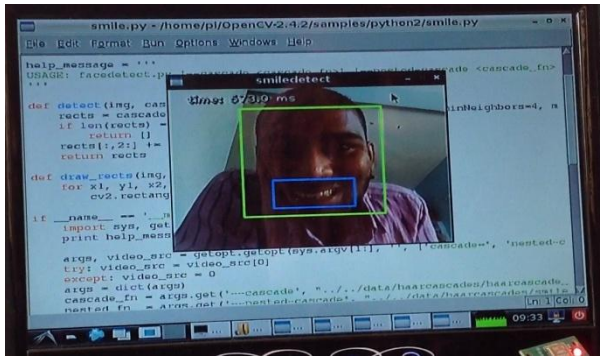


Fig 7: Smile detection shown by blue rectangle when person posed smile.

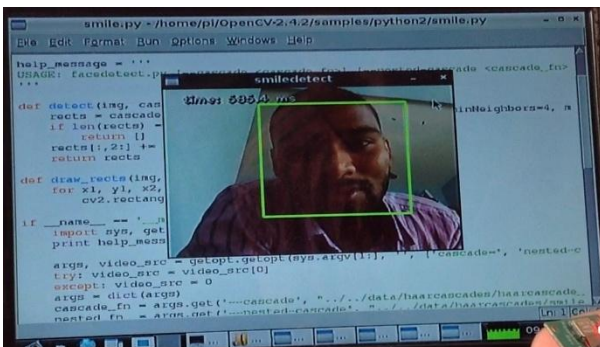


Fig 8: Smile not detected when person not posed smile. Blue rectangle not appeared at mouth region.

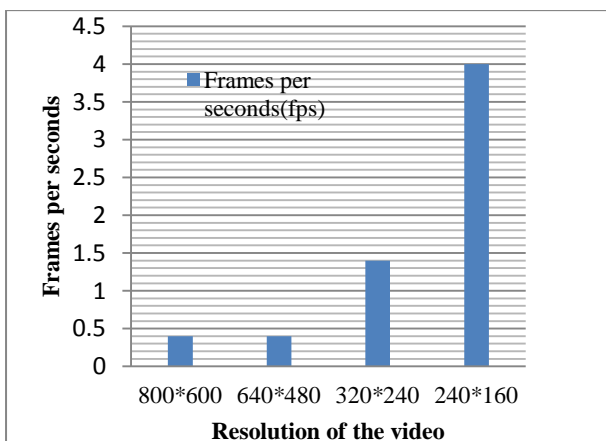


Fig 9: Experimental graphical results of various video resolutions with low frame rate when smile detection takes place.

7. CONCLUSION

Smile detection in real life faces is being challenging for many researchers in real time video with small and low cost hardware. Smile detection in face images captured in real-world scenarios is an interesting problem with many applications. With advancement real time smile detection using raspberry pi leads to motivation to students and researchers to build the small low cost robots that work with

facial expressions and human computer interfaces applications. By the experimental results states that the low cost raspberry pi device can able detect smile in real life environments and it can be usable in practical applications with low cost approach.

8. REFERENCES

- [1] Z. Zeng, Y. Fu, G. I. Roisman, Z. Wen, Y. Hu, and T. S. Huang, "Spontaneous emotional facial expression detection," *J. Multimedia*, vol. 1, no. 5, pp. 1–8, Aug. 2006.
- [2] J. Whitehill, G. Littlewort, I. Fasel, M. Bartlett, and J. Movellan, "Towards practical smile detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 11, pp. 2106–2111, Nov. 2009.
- [3] Caifeng Shan Member, *IEEE* "Smile Detection by Boosting Pixel Differences," *IEEE Trans. on Image processing*, vol. 21, No 1, January 2012.
- [4] Yen Chang, "Low cost real time face detection, tracking and recognition for human robot interactions" https://etd.ohiolink.edu/etd.send_file?accession=case1307548707&disposition=inline
- [5] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2001, pp. 511–518.
- [6] R. E. Schapire, "The boosting approach to machine learning: An overview," in *MSRI Workshop Nonlinear Estimation Classification*, 2002, pp. 1134–1227.
- [7] C. Yoon, M. Cheon, E. Kim, M. Park, and H. Lee, "Real-time road sign detection using adaboost and multicandidate," in *Proc. 8th Symp. Adv. Intel. Syst. (ISIS)*, 2007, pp. 953–956.
- [8] G. Donato, M. Bartlett, J. Hager, P. Ekman, and T. Sejnowski, "Classifying facial actions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 10, pp. 974–989, Oct. 1999.
- [9] Daniel Devatman Hromada, prof. Charles Tijus, "Semi supervised haartraining of a fast&frugal open source zygomatic smile detector," *IEEE Int. Conf. computing and communications RIVF 2010*.
- [10] Open Computer Vision Library [Electronic resource], <http://sourceforge.net/projects/opencvlibrary/files/opencv.2012>
- [11] M. Richardson and S. Wallace, "Getting Started with Raspberry Pi", 1st ed. O'Reilly, Media, U.S.A: 2012,
- [12] R. P. Website. The making of pi, 2014. <http://www.raspberrypi.org/about/>.
- [13] Raspberry Pi camera module setup www.raspberrypi.org/help/camera-module-setup/