THE ANNALS OF "DUNĂREA DE JOS" UNIVERSITY OF GALATI
FASCICLE III, 2022, VOL. 45, NO. 1, ISSN 2344-4738, ISSN-L 1221-454X
ELECTROTECHNICS, ELECTRONICS, AUTOMATIC CONTROL, INFORMATICS
Article DOI: https://doi.org/10.35219/eeaci.2022.1.03

ON-LINE VIDEO PROCESSING EMBEDDED SYSTEM FOR AUTOMOTIVE

G. Petrea, V. Nicolau, M. Andrei

Department of Electronics &Telecommunications,
"Dunărea de Jos" University of Galati, Romania.
george.petrea.@ugal.ro, viorel.nicolau@ugal.ro, mihaela.andrei@ugal.ro

Abstract: Real-time processing for images is increasingly used in automotive industry. Video signals are nowadays used for different applications on traffic, such as measuring average speed, monitoring traffic congestion, detect red light crossing, but also for sign recognition, lane assist etc. The paper proposes an embedded system for image processing used for controlling climatization in cars. This system is also useful in case of an accident to activate the airbag if a person is on the seat or not to do it if a heavier object is placed on the seat, for example a bag. For this aim, a Raspberry Pi3 based system is proposed for real-time detecting of the number of passengers existing in a vehicle. The Raspberry Pi3 is used for acquiring frames from a video camera connected via USB interface and for on-line processing of images. The system uses an Haar cascade classifier algorithm (HCCA) for human face detection.

Keywords: image processing, automotive, DSP, real time.

1. INTRODUCTION

Video processing for object detection involves the extraction of those visual features with well-defined properties such as accuracy and stability for obtaining reliable results in terms of time of processing and precision.

Different types of face detection methods can be used, but four general categories can be defined: feature based method, that locates faces using structural features that are extracted from the images, knowledge based method, that uses human knowledge about face detection and depend on a hard to define set of rules about distances or relative positions of face components, template matching method, that uses predefined templates for face detection by correlating them to the input image and appearance based method, that uses statistical analysis and also machine learning techniques in order to find relevant parameters of face images. The appearance-based method is the best solution for face detection and different algorithms have been used in time for face detection.

For face detection different approaches were made during time. For example, (Rowley et al., 1996) presents neural networks based detection system. The main idea used for reducing false recognitions is that faces should be detected by various trained neural networks. Moreover, more than one positive response should be obtained in various nearby rectangles.

In (Kotropoulos and Pitas, 1997) a face localization system is implemented based on proposed a hierarchical rule. The idea was to scan the original image according to a well-known set of rules for a 6x7 pixels rectangle. This procedure is repeated for changing image resolutions. Different rules are also applied for distinguishing the eyes, eyebrows and mouth position.

In the article presented by (Viola and Jones, 2001) is the first time introduced the idea of using Haar cascade classifiers for detecting human faces. The approach is to create a serial chain of boosted classifiers which can ensure reliable and fast face detection. The same idea was continued in (Lienhart et al., 2002) but with the improvement of using rotated features together with the initial ones.

In this paper the object proposed to be detected is the human face. A Haar cascade classifier algorithm (HCCA) is used for serially identifying the head, the eyes, nose and mouth of people in an image extracted from a video input. The HCCA is implemented in a Raspberry Pi3. The next of the paper is structured as it follows: In section 2 are presented some basics on

Haar cascade classifiers. Section 3 presents the proposed system for face counting in a vehicle. Section 4 shows some results and Section 5 conclusions.

2. BASE OF HAAR CASCADE CLASIFIER

The Haar algorithm is basically a full picture crossing by different defined features. Finally, the classifier resulted notes positive data points if there are part of the detected object (human face in our case), or negative data point.

Firstly, defined in (Viola and Jones, 2001), the detector is a good combination of three main components. Initially, a set of features with the property that can be processed in a short and relatively constant period of time are defined. The idea of this first approach is that it can help by increasing the differences between classes and reducing the differences inside one class.

The next step is to apply an enhanced algorithm that will help the selection of the most important features in the same time with the training of the classifier.

The last step in the algorithm is to create a cascade series of classifiers with increasing complexity for completing the detection procedure.

The Haar feature can be defined as a detection window having WxH pixels using the equation:

(1) $$feat=\sum_{i=1}^{N}\omega_i\cdot RSum(r_i)$$

In equation (1) ω_i is a weighting factor that is randomnly chosen and $RSum(r_i)$ is a sum of the intensity values of any kind of rectangle (vertical or rotated) r_i that is placed in the detection window.

In images, a rectangle can be completely defined using five parameters: $r=(x,y,w,h,\theta)$, where x and y represent the coordinates of the top left corner of the rectangle, w and h are the rectangle dimensions and θ gives the rotation angle.

Equation (1) gives thus an almost unlimited variety of features. Hence, some new restrictions must be applied:

- There are allowed only pixel sums for only two rectangles ($N=2$).
- The weights are used for compensating the area difference of those two rectangles and will have different sings. This will imply that $-\omega_1\cdot Area(r_1)=\omega_2\cdot Area(r_2)$.
- The chosen features should be equivalent to the ones used in the preliminary stages of human vision.

After these restrictions are applied remain only 14 basic features which have the property that can be used in every part of the detection window and can be rotated and scaled in both directions. Figure 1 presents the basic features.

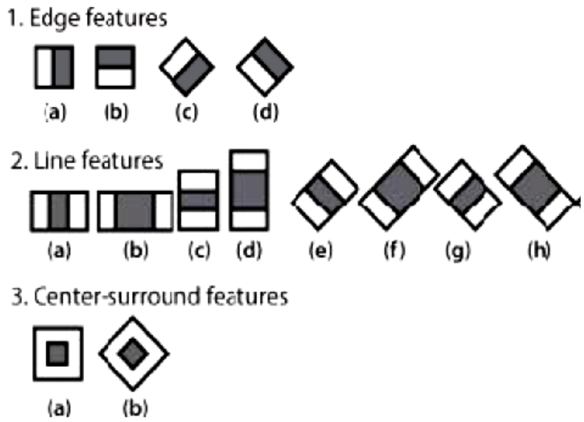

Fig.1. Haar basic features used for face detection.

In these cases, the features computation uses the black rectangle for determining the sum of difference of pixel intensity and the white rectangle for scaling in order to compensate the differences of the areas.

The object (face in this case) uses a small part of the image. The cascade classifier permits to solve the detection problem fast, by first discarding the regions that are not of interest and then focusing on the most relevant parts.

The idea is to have N stages of classifiers that are connected in a serial way, as in figure 2. Each classifier will decide on a specific feature if it is part of the detected object (face) or not.

For each stage of processing a window are obtained ratios for the case of true positive (TP) of at least p and false positive ratios (FP) of at most f. After the computation of the ratios, only the windows with positive ratios go through the next step and for the other windows the operation stops, thus reducing processing time.

The overall detection ratios for all the N stages will be exponential functions of the ratios computed for a single stage:

(2) $$TP_{total}=\prod_{i=1}^{N}p_i\approx p^N.$$

$$FP_{total} = \prod_{i=1}^{N} f_i \approx f^N \quad (3)$$

where TP_{total} and FP_{total} are true and false ratios of the entire cascade.

By correctly choosing the p and f ratios (usually p close to 1 and f usually 0.5) for the N stage cascade it will result a detector that will maintain a high ratio for true positive cases (close to 100%) together with and a low ratio for false positive cases (close to 0%).

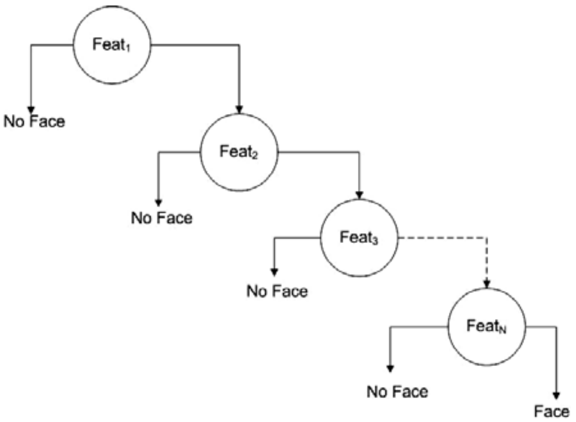

Fig.2. Principle of HCCA for face detection

3. DEVELOPMENT OF PROPOSED SYSTEM

The embedded system should be capable of real time image processing for determining the number of passengers in a vehicle. For this purpose, various solutions were discussed but finally the system implemented is based on a Raspberry Pi3 platform together with a wide-angle USB video camera.

The camera chosen is Logitech C920 Pro HD Webcam which is a 3-megapixel webcam with a maximum resolution of 1080p/30fps - 720p/30fps. Also of interest in this project is the diagonal field of view which in this case is of 78°. Other specifications of the camera worth mentioning are autofocus capabilities and built-in noise-cancelling microphone that can be used in other applications.

The Raspberry Pi3 platform comes with a Quad Core 1.2 GHz Broadcom BCM2837 64bit CPU, 1GB RAM, BCM43438 wireless LAN and Bluetooth Low Energy (BLE) on board for wireless external communication, 100Base Ethernet for cabled interconnection, 40-pin extended GPIO for communication with the Climate Controller of the car or for other general use sensors, 4 USB2 ports used for webcam connection. Other characteristics which worth mentioning but not used in this project are: 4 Pole stereo output and composite video port, CSI camera port for connecting a Raspberry Pi camera, DSI display port for connecting a Raspberry Pi touchscreen display. For widespread use such as programming the platform, Raspberry Pi3 is also equipped with full size HDMI, Micro SD port for loading the operating system and storing data and upgraded switched Micro USB power source up to 2.5A.

The algorithm for face detection is implemented in the Raspberry Pi3 single board computer platform, which provides enough computing power for this kind of application. Figure 3 shows the embedded system proposed for face detection. Basically, a USB web camera is needed and the Raspberry Pi3 platform will be connected to the Climate Controller of the car, using one of the integrated serial communications.

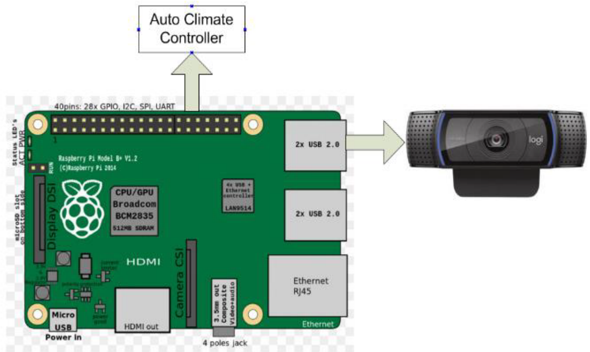

Fig.3. Video embedded system proposed

RGB to grayscale operation needs to be performed first in order to reduce the complexity of the algorithm. After that, some enhancements are applied to the image, such as resizing or sharpening. Next, segmentation needs to be done, to detect different objects in the image. These are the preliminary steps before applying the HCCA algorithm.

This algorithm is based on the idea of the universal properties of a human face, such as: nose region is usually brighter than the eye region or the eyes region is darker than the pixels next to it etc. Using line, edge and center detection, the algorithm identifies if the picture in work contains eyes, nose, mouth and establishes if a face is detected.

The HCCA as well as the rest of the steps in the real time processing is implemented using Python programming language on the Raspberry Pi platform, using 4 features as shown in figure 2.

For the first feature the classifier detects human heads or objects resembling to a human head. The rest of the image is not further processed. For the second step the classifier searches for eyes in the group of objects resembling to a human face. Also, at this step the rest of the objects not containing eyes are not furthermore processed.

Next, for the third step the HCCA searches in the remained object if a nose is present and positioned in the face. Finally, the algorithm searches the remained object for mouth detection and if the result is true the face is marked with a rectangle in the picture as shown in figure 5.

4. RESULTS AND FUTURE WORKS

The algorithm described above is an on-line processing used in automotive climate car control. The diagram presented in figure 4 shows the main objectives of the software implemented in the Raspberry Pi3.

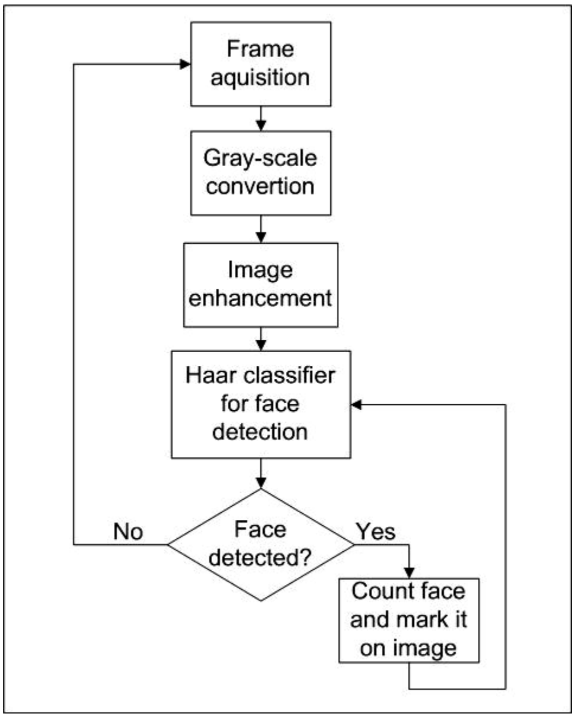

Fig.4. Algorithm diagram for face counting

Being an on-line embedded system, processing time is a key parameter in this project. After setting the features for the classifiers, as described in the above paragraph, the next step is to choose the optimum resolution for counting the people in a vehicle and for obtaining the best time for processing. Some resolutions have been tested and in table 1 are shown the resulted time of processing the HCCA.

Table 1 Processing periods per frame

Image Resolution	***Bytes***	**Processing time per frame (ms)**
800x600	180000	0.25
640x480	115200	0.14
320x240	28800	0,078

Figure 5 shows a sample of the resulting image obtained for face detection. Further the system transmits only the counted faces to the Climate Control system.

For future works the idea of face detection for different head positions will be taken into account in order to improve the detection ratio of the system. Also, because of the tight space inside the car, the idea of 2 cameras used for face detection could be implemented.

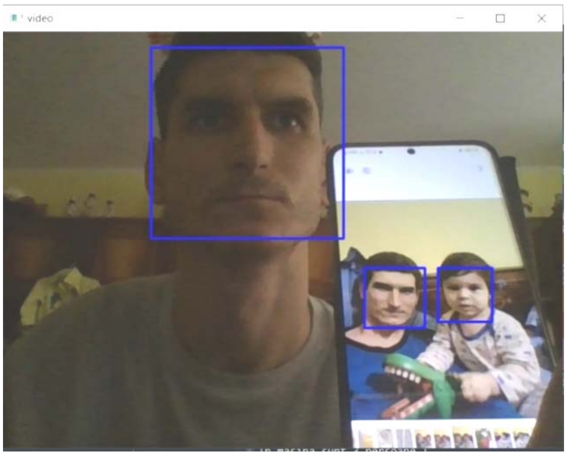

Fig.5. Human face detection with proposed system

5. CONCLUSIONS

This paper proposes an on-line video processing embedded system for human face detecting and counting. This system is to be used in automotive for controlling the Climatic Control system of a vehicle.

Different solutions have been studied, but HCCA for human face detected has been chosen and implemented using Raspberry Pi3 platform and wide-angle video camera. The time processing has been monitored due to the importance of this parameter regarding real time system and suitable resolution and features have been chosen for implementing the algorithm.

REFERENCES

Badura, S., Foltan, S.: *Advanced scale-space, invariant, low detailed feature recognition from images—car brand recognition.* In: Computer Science and Information Technology (IMCSIT), Proceedings of the 2010 International Multiconference on, pp. 19–23 (2010). doi:10.1109/IMCSIT.2010.5679924

Gonzalez R C, Woods R E., *Digital Image Processing (3rd Edition)*, Prentice Hall, 2008, New Jersey 07458, USA.

Kaehler A, Bradski G, (2016) *Learning OpenCV 3*, O'reilly Media, Inc., Sebastopol, CA 95472, USA.

Kotropoulos C, Pitas I (1997) Rule-based face detection in frontal views. In: Proceedings of the international conference on acoustics, speech and signal processing 1997, pp 2537–2540

Lienhart R, Kuranov A, Pisarevsky V (2002) *Empirical analysis of detection cascades of boosted classifiers for rapid object detection*. Intel Labs, Microprocessor Research Lab Technical report

Mandal B., *Face recognition: Perspectives from the real world*, IEEE 2016 14th International Conference on Control, Automation, Robotics and Vision (ICARCV), pp. 1-5, 13-15 Nov. 2016, Phuket, Thailand.

Nagabhushana S., *"Introduction," in Computer Vision and Image Processing*, New Age International Ltd., Publishers, 2005, p.3.

Rowley H, Kanade T, Baluja S (1996) *Neural network-based face detection.* In: Proceedings of the IEEE conference on computer vision and pattern recognition 1996, pp 203–207

Viola P, Jones M (2001), *Rapid object detection using a boosted cascade of simple features.* In: Proceedings of the IEEE conference on computer vision and pattern recognition 2001, pp 511–518).

Wilson P., Fernandez J. (2006), *Facial features detection using Haar classifiers.* J Comput Sci Coll 21:127–133

Zhang K, Zhang Z, Li Z., (2016) *Joint face detection and alignment using multitask cascaded convolutional networks,* IEEE signal Processing Letters, 2016, 23(10):1499-1503.