Embedded Face Recognition System Based on Multi-task Convolutional Neural Network and LBP Features

Mengyue Zhang[1,a], Weihan Liao[1,b], Jianlian Zhang[1,c], Huisheng Gao[1,d], Fanyi Wang[1,e], Bin Lin*[1,f]

[1]Zhejiang University, State Key Laboratory of Modern Optical Instrumentation, College of Optical Science and Engineering
Hangzhou, China
E-mail: [a]3130101841@zju.edu.cn, [b]cliaoweihan@foxmail.com, [c]zhangjianlian@zju.edu.cn, [d]gaohuisheng@zju.edu.cn,
[e]11730038@zju.edu.cn, [f]wjlin@zju.edu.cn

Abstract

Based on neural network and local binary pattern algorithm, this paper builds a lightweight artificial face recognition system on chip Firefly-RK3399, with high speed, strong robustness and high recognition accuracy. Our embedded artificial intelligent face recognition system mainly consists of face detection, feature extraction and recognition. Multi-task convolutional neural network (MTCNN) under the CaffeOnACL framework is utilized for face detection, and the local binary pattern (LBP) is applied as face recognition algorithm. Experiments illustrate that our artificial intelligent embedded face recognition system has high speed and accuracy, which is easy-carrying and of high commercial value as well.

Key words: face recognition; multi-task conventional neural network; local binary pattern; embedded system

1. Introduction

Face recognition has always been one of the most challenging tasks in the field of target recognition. Meanwhile, as one of the most effective ways of personal identity verification in modern society, face recognition has important application value. It has been extensively used in monitoring, security, communication, human-computer interaction and other fields [1]. Therefore, efficient and accurate algorithms are especially significant for face recognition in limited embedded system resources such as mobile phones and chips. In recent years, there have been more and more researches on the field of face recognition algorithms applying on embedded platform [2].

With the rapid development of artificial intelligence, and its wide application in face recognition, how to integrate artificial neural network into an embedded face recognition system comes to be a new topic [3].

Taking the speed, stability and accuracy of face recognition system in an embedded system into consideration, a combination of traditional feature recognition algorithm based on LBP and deep learning algorithm based on MTCNN is proposed in this paper to meet these requirements in a certain degree.

2. Algorithm Based on MTCNN and LBP

In order to ensure that face recognition algorithm can be transplanted to capacity limited embedded system and meanwhile guarantee a high efficiency, we develop a face recognition algorithm combining neural networks and traditional LBP features. As shown in Fig.1, our recognition algorithm consists of image acquisition, image preprocessing, face detection and alignment based on MTCNN, feature extraction based on LBP, and face recognition with logic optimization for multi-objective. They will be introduced in detail following.

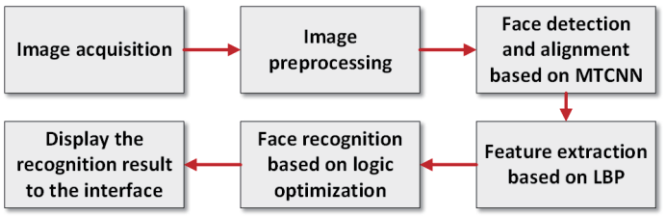

Fig.1 Flow chart of embedded face recognition system.

A. Face Detection and Alignment based on MTCNN

In face recognition system, face detection and alignment are two important steps. For these two steps, we apply MTCNN to increase the accuracy and stability of system.

In common situation, various poses of head, illuminations of the environment will affect the stability of face recognition system. Researches have shown that compared with traditional algorithms, deep learning methods have better performance on face detection and alignment [4].

MTCNN is essentially a cascading architecture to integrate the problem of multi-task CNNs learning. Under a deep cascading multitask framework which makes full use of the inherent relationship between detection and alignment, the performance of networks is greatly increased. Through a third-order concatenated convolutional neural network, the network separates the task of face detection and alignment from coarse to fine processing, dividing into three parts: face classification, bounding box regression, and facial landmark localization.

The whole architecture contains three steps which is shown in Fig.2 [5]. The first step, a shallow CNN quickly generates candidate facial windows and conduct non-maximum suppression to remove overlapped windows; the second step, through more complex CNNs, refining candidate facial windows, and discarding a large number of overlapped facial windows again through NMS; the third step, using more powerful CNNs, selecting out which windows to keep and meanwhile displaying five key points in faces. The network at each step is a multitasking network.

ISBN: 978-1-5386-8139-8

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 15,2020 at 01:30:45 UTC from IEEE Xplore. Restrictions apply.

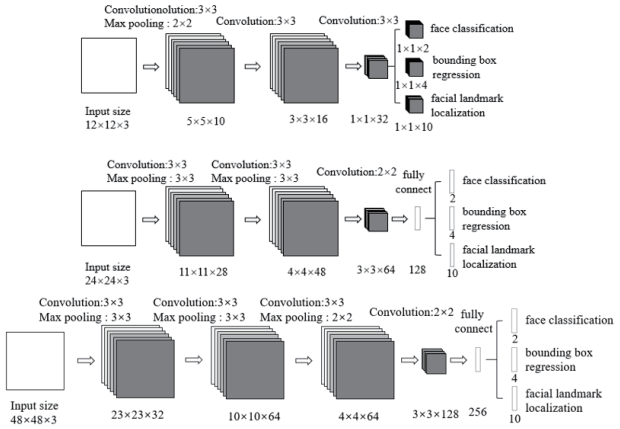

Fig.2 The architecture of MTCNN.

In the process of training, the loss functions are determined by different tasks. Concretely, the loss function of face classification is cross-entropy, expressed as (1), and the loss function of bounding box regression is Euclidean loss, expressed as (2).

$$L_i = -\left(y_i \log(p_i) + (1-y_i)(1-\log(p_i))\right) \quad (1)$$

where p_i is the probability of detected target being a human face, and y_i is 0 or 1, a ground-truth label.

$$L_i = \left\| y_i - y_i \right\|_2^2 \quad (2)$$

where y_i is each facial landmark's coordinates, and y_i is the ground-truth coordinate for i-th sample.

After training the three parts of the network separately with mentioned loss functions, we get suitable parameters of the networks. Meanwhile, the whole capacity of our convolutional neural network is within an allowable range to be applied in our embedded system.

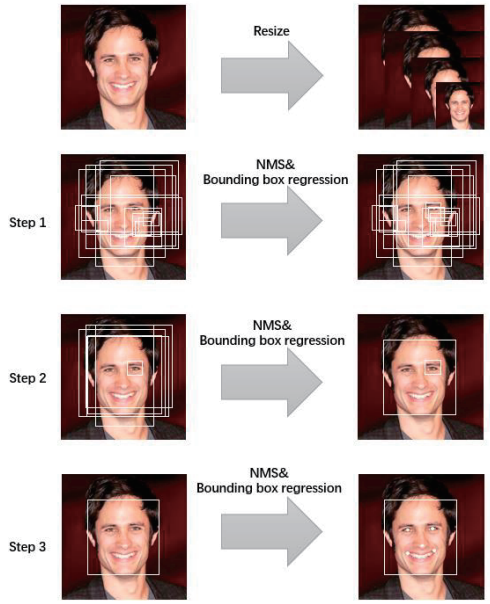

Fig.3 The three steps of face detection and alignment by MTCNN.

B. Feature Extraction and Recognition based on LBP

The extraction of LBP feature is fast, and it is widely applied in face feature extraction.

The original LBP operator is defined as following: in a 3*3 window, the gray level of 8 adjacent surrounding pixels is compared with the center pixel as a threshold, and if the gray level of the surrounding pixel is bigger than or equal to the central pixel, then the pixel is marked as 1, otherwise marked as 0, as shown in Fig.3.

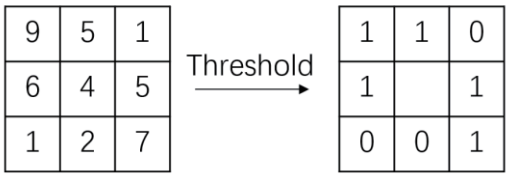

Fig.3 Process of original LBP operator in a 3*3 window.

For the detected face image, we perform binarization processing and LBP feature extraction. During the processing, we select a circular LBP operator. The radius of the operator is 1 and the number is 8. Considering the rotation invariant mode of LBP, as shown in Fig.4, we divide the binary image to 8*8 blocks, and then count the number of each LBP pixel value in each small block horizontally and vertically in order.

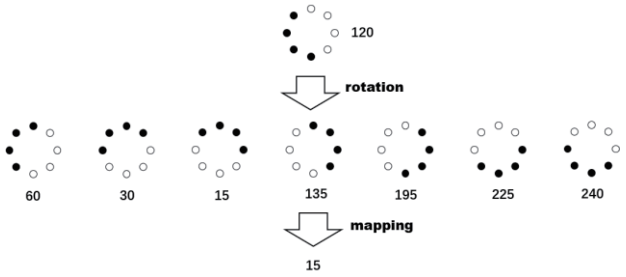

Fig.4 Rotation invariance of LBP

According to the position of important features in the face, we distribute different weights of each small block in face image, as shown in Fig.5. Then obtain the feature vector of the image.

0	0	0	0	0	0	0	0
0	1	2	5	2	1	0	0
1	5	15	20	15	5	1	0
2	15	35	30	35	15	2	0
4	10	35	30	35	10	4	0
2	5	10	25	10	5	2	0
1	3	8	20	8	3	1	0
0	2	6	6	6	2	0	0

Fig.5 Different weights of each small block in face image.

An LBP operator can produce different binary patterns. For

ISBN: 978-1-5386-8139-8

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 15,2020 at 01:30:45 UTC from IEEE Xplore. Restrictions apply.

LBP operators with P samples in a circular area of radius R, 2P kinds of modes will be generated. Obviously, as the number of sampling spots in the domain set increases, the type of binary pattern increases dramatically. Ojala defines the "equivalent mode" of LBP as: when a loop binary number of a local binary mode has a maximum of two times of transitions from 0 to 1 or from 1 to 0, it is reserved as an equivalent mode class, and for situations more than two times, we classify them into another class [6].

Under this mode, the number of binary mode types is greatly reduced, from 2^P to $P(P\text{-}1)+2+1$, where P is the number of sampling spots in the domain set, and the equivalent mode class contains $P(P\text{-}1)+2+1$ modes, mixed mode class has only 1 mode. This makes the feature vector less dimensional and can reduce the influence caused by high frequency noise.

Considering the equivalent model of LBP, there are 59 different LBP values, and the length of the feature vector above is 59*8*8.

The chi-square is used to compare the similarity of two face feature vectors. In LBP face recognition, the image is divided into N blocks with the same size. In these small blocks, the histogram features are extracted by LBP, so that each image has a lot of histogram information, and the histogram of the image to be matched is $S_{i,j}$, the histogram of the known image is $M_{i,j}$, where $i=1,2,3...,N$ is a small area of the image, j is the value of a column of the histogram in the small area, then the similarity of two different images is:

$$\chi_{\omega}^{2}(S,M)=\sum_{i,j}\omega_j\frac{(S_{i,j}-M_{i,j})^2}{S_{i,j}+M_{i,j}} \tag{3}$$

where ω_j is the weight of each small block. For example, in the face image, the area of the eyes and mouth contains more information, so the weight of these areas can be set larger. The denominator part of the above formula is $S_{i,j}+M_{i,j}$, considering the difference of the same face in different photos. The smaller the comparison value of the final output, the higher the similarity.

3. Logic Optimization for Multi-objective Recognition

The feature vector of a new face image is calculated and stored with a label when registering. In the process of face recognition, the most similar face is found by comparing the face feature vector of current frame with those of the registered faces in our face database.

For the case where multiple faces appear in the scene at the same time, the traditional algorithm compares each face with the database on each frame, and finally outputs the label with the largest similarity as the recognition result. However, this may lead to errors in some frames, which is, one's face may change largely and therefore be recognized as another label or even a determined label in the scene.

Considering the case where there are N classes in the face database and M faces in the camera detection screen, firstly, for each face, we calculate the similarity with each class in database, and store these similarities in a matrix of $N*M$. Then, we find the highest similarity element in the matrix, and determine that the corresponding row and column of the element are the same person. Then exclude the corresponding row and column and then look for other element which has highest similarity, and so on, until we match all the classes in the database with the faces in the scene.

This strategy can improve the efficiency and reliability of face recognition under multi-objective recognition.

4. Implementing on Embedded Platform

Firefly-RK3399 is a high-performance open source board built by Firefly, with 6 cores, clocked at $2.0GHz$, 4G DDR4 dual Channel 64-bit RAM, and has rich interfaces and stable performance.

OpenCV is an open-source computer vision library initiated and developed by Intel Corporation [7]. To implement the algorithm on this embedded system, we remove high-gui module of OpenCV and cross-compile it on the X86 platform.

In the experiment, the face recognition algorithm described above based on MTCNN and LBP features is implemented on Firefly-RK3399. When the recognition target is a single face, it takes about $120ms$ to complete the process of detecting and identifying, and the average recognition speed is about 8 frames per second. When the recognition target is three faces simultaneously, it takes about $200ms$ to complete the process, and the average recognition speed is about 5 frames per second.

Further experiments illustrate that when there are 58 users in face database, our system can correctly identify 56 out of them, and the recognition rate is 96.6%. Meanwhile, when detecting 12 faces that are out of face database, our system can figure out that they are not registered users.

Fig.6 The result of recognizing three different faces simultaneously.

5. Conclusion

The experiments illustrate that our face recognition system based on MTCNN and LBP features can achieve high recognition rate on embedded devices in limited data. Neural network greatly optimizes the accuracy and stability of the system. Meanwhile, the high extraction speed of LBP further increases the speed of system. The recognition rate of our system

ISBN: 978-1-5386-8139-8

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 15,2020 at 01:30:45 UTC from IEEE Xplore. Restrictions apply.

is 96.6% under the circumstance of there are 58 users in face database, and the recognition speed is 8 *fps* when recognizing a single face and 5 *fps* when recognizing three faces simultaneously. Obviously, our embedded face recognition system has advantages of easy carrying, high speed and accuracy.

References

[1] Li S, Xin L, Chai X, et al. Morphable Displacement Field Based Image Matching for Face Recognition across Pose[C] European Conference on Computer Vision. 2012.

[2] Aby P K, Jose A, Jose B, et al. Implementation and optimization of embedded Face Detection system[C] International Conference on Signal Processing. 2011.

[3] Dong W, Jing Y, Deng J, et al. FaceHunter : A multi-task convolutional neural network based face detector[J]. Signal Processing Image Communication, 2016, 47:476-481.

[4] Hu G, Yang Y, Dong Y, et al. When Face Recognition Meets with Deep Learning: An Evaluation of Convolutional Neural Networks for Face Recognition[J]. 2015.

[5] Zhang K , Zhang Z , Li Z , et al. Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks[J]. IEEE Signal Processing Letters, 2016, 23(10):1499-1503.

[6] Ojala T, Pietikäinen M, Mäenpää T. A Generalized Local Binary Pattern Operator for Multiresolution Gray Scale and Rotation Invariant Texture Classification[J]. 2001.

[7] Guennouni S, Ahaitouf A, Mansouri A. Multiple object detection using OpenCV on an embedded platform[C] Information Science & Technology. 2015.

ISBN: 978-1-5386-8139-8

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 15,2020 at 01:30:45 UTC from IEEE Xplore. Restrictions apply.