JOURNAL FOR INNOVATIVE DEVELOPMENT IN PHARMACEUTICAL AND TECHNICAL SCIENCE

Volume-2,Issue-6 (June-2019)
ISSN (O) :- 2581-6934

Face Detection and Face Recognition using Face Segmentation Technique

G. Abisha Vincent [1] , G.G.Godwin Raj [2]

[1 2] Student, C.S.I. Institute of Technology

Abstract :- Methods for segmenting face and for detecting facial landmark is presented. A face recognition which is automatic and the input is just the information of the depth of the face is the aim of development. The whole process of segmentation includes edge detection, region clustering, shape analysis is also included to extract the face region. The addition of detection of a surface curvature information and depth relief curve greatly help in finding the nose and eye land marks to evaluate the proposed system and its edges process for 3D recognition uses. The accuracy of the segmentation and landmark detection are analyzed. It has a better edge over the state of the arc works. The work of segmentation is a 3D recognition system where evaluated and to improvements obtained by application of landmark based technics where analyzed.

Introduction

The term Image processing in general refers to the processing of two dimensional picture by a digital computer. In a wider context the image processing means digital processing of the two-dimensional matrix kind of data. Out of five senses of human, the “sight” is the most powerful. Receiving and analyzing images forms a large part of the routine cerebral activity of human beings.in fact, more than 99% of the activity of the human brain is involved in processing images from the visual cortex. A visual image is rich in information. Confucius said. ” A picture is worth a thousand words”. On a more sophisticated level, human generate, record and transmit images. Images can be formed from many kinds of objects using differing mechanisms of formation. The image system classifies according to the type of radiation or field used to form an image. Electromagnetic radiation is used most often in imaging systems. The radio frequency band is used in astronomy and in magnetic resonance imaging (MRI). Microwaves are used in radar imaging since they can penetrate clouds and other atmospheric conditions that interface with imaging using visible light. The advent of computers has opened up vast new possibilities for the quantitative processing and analysis of images, as long as these can be represented by arrays of discrete values, rather than continuous functions. In the case of analog images, they can be converted into digital images by a two-step process sampling and quantization known as digitization. Some of the Image processing functions are image enhancement, brightness adjustment, contrast enhancement, Image averaging, convolution, frequency domain filtering, edge enhancement, image restoration, photometric correction, inverse filtering, image analysis, segmentation, feature extraction, object classification, image compression lossless and lossycompression, image synthesis, tomographic imaging,3-D reconstruction image and etc.,

1.1 Overview of Project

The goal was to develop an automatic process to be embedded in a face recognition system using only depth information as input. Facial landmark detection that combines surface curvature classification and depth relief.

1.2 Problem Description

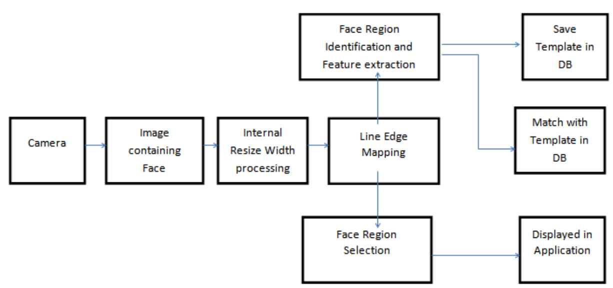
The recognition of fear expression appears to be the most challenging in SIM technique and the effect of pose distortion on 3D facial data is not well studied. The faces with shape variation is difficult to find in the Frenet frame and also the verification rating is very low. In matching the detection of landmark is complicated.

1.2.1 Need for Face Segmentation

With the development of 3-D face recognition techniques, face segmentation and landmark detection on depth information have become very important preprocessing steps for designing fully automatic recognition systems. The need for face segmentation was observed in some of the methods aiming for 3-D face recognition, which were based on different strategies (e.g., image registration, appearance recognition, and profile curves. In all these works, the face was considered as a rigid shape, and the same drawback was observed: recognizing faces with shape variations. It was found that even small changes between two neutral expression faces of the same subject could cause a decrease in the verification rate. Subsequent works in the literature successfully employed landmark detection to better deal with this problem. Change proposed the use of three different regions around the nose during the matching stage proposed a modeling approach for expression deformation.

Bronstein Proposed an expression-invariant face representation for recognition. Surface curvature classification has been successfully applied for landmark detection showing high localization rates. Employed curvature information together with some heuristics to find facial landmarks in face images with arbitrary pose. However, this required a large displacement tolerance to obtain good localization rates. The profile curves from different views of the input image to detect the nose tip in non- frontal face images. It obtained high localization rates considering a small displacement tolerance is restricted to a single landmark. For frontal face images improved precision and detection rates by combining surface curvature classification with relief curves of depth images, statistical model of landmark location, and point cornerness obtained from intensity images. In this method, a training stage was required to create a statistical model, and the use of intensity images may present some problems regarding illumination, pose, and alignment between intensity and range data. To perform landmark detection, each range image must be first segmented seeking the isolation of the face region from the other parts (i.e., regions that do not belong to the face area) such as hair, neck, ears, and clothes. For images containing only one subject, many techniques have been adopted for face segmentation, like data clustering skin color models and histogram of depth coordinates and even manual segmentation has been reported. Some of these methods use the color information to perform segmentation and are subject to the same problems that arise when using intensity images. Other works can detect more than one face per image either by creating 2-D projections of the 3-D data and applying appearance training methods, or performing techniques based on a boosted cascade classifier that uses color information to establish the face location from the range data. It doesn't extract the entire face region. Only the area containing the nose and the eyes is extracted, and in the face segmentation technique excluded forehead and chin parts. We present a methodology for face segmentation and facial landmark detection in range images. Our goal was to develop an automatic process to be embedded in a face recognition system using only depth information as input.

SYSTEM ARCHITECTURE:



The input image should obey the following constraints.

- ❖ The image cannot contain more than one face
- ❖ The image must be among the closest region to the acquisition device
- ❖ The face must be frontally posed
- ❖ The face cannot be occluded by but except hair

The input image should satisfy the above constraints to segment the face first of all we need to isolate face region from the background.

ISSN:-2581-6934|www.jidps.com

Algorithm

```
Algorithm: Viola-Jones Face Detection Algorithm
Input: original test image
Output: image with face indicators as rectangles
for i ← 1 to num of scales in pyramid of images do
    Downsample image to create image_i
    Compute integral image, image_ii
    for j ← 1 to num of shift steps of sub-window do
        for k ← 1 to num of stages in cascade classifier do
            for l ← 1 to num of filters of stage k do
                Filter detection sub-window
                Accumulate filter outputs
            end for
            if accumulation fails per-stage threshold then
                Reject sub-window as face
                Break this k for loop
            end if
        end for
        if sub-window passed all per-stage checks then
            Accept this sub-window as a face
        end if
    end for
end for
```

The Viola-Jones algorithm is a widely used mechanism for object detection. The main property of this algorithm is that training is slow, but detection is fast. This algorithm uses Haar basis feature filters, so it does not use multiplications.

The efficiency of the Viola-Jones algorithm can be significantly increased by first generating the integral image.

$$II(y,x) = \sum_{p=0}^{y} \sum_{q=0}^{x} Y(p,q)$$

The integral image allows integrals for the Haar extractors to be calculated by adding only four numbers. For example, the image integral of area ABCD (Fig.1) is calculated as $II(y_A,x_A) - II(y_B,x_B) - II(y_C,x_C) + II(y_D,x_D)$.

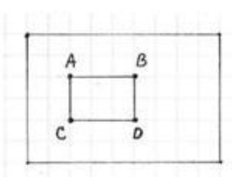


Fig.1 Image area integration using integral image

Detection happens inside a detection window. A minimum and maximum window size is chosen, and for each size a sliding step size is chosen. Then the detection window is moved across the image as follows:

1. Set the minimum window size, and sliding step corresponding to that size.
2. For the chosen window size, slide the window vertically and horizontally with the same step. At each step, a set of *N* face recognition filters is applied. If one filter gives a positive answer, the face is detected in the current widow.
3. If the size of the window is the maximum size stop the procedure. Otherwise increase the size of the window and corresponding sliding step to the next chosen size and go to the step 2.

Each face recognition filter (from the set of *N* filters) contains a set of cascade-connected classifiers. Each classifier looks at a rectangular subset of the detection window and determines if it looks like a face. If it does, the next classifier is applied. If all classifiers give a positive answer, the filter gives a positive answer and the face is recognized. Otherwise the next filter in the set of *N* filters is run.

Each classifier is composed of Haar feature extractors (weak classifiers). Each Haar feature is the weighted sum of 2-D integrals of small rectangular areas attached to each other. The weights may take values ±1. Fig.2 shows examples of Haar features relative to the enclosing detection window. Gray areas have a positive weight and white areas have a negative weight. Haar feature extractors are scaled with respect to the detection window size.

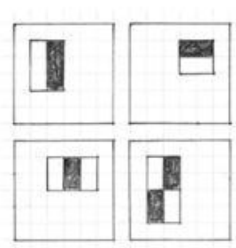


Fig.2 Example rectangle features shown relative to the enclosing detection window

The classifier decision is defined as:

$$C_m = \begin{cases} 1, & \sum_{i=0}^{l_m-1} F_{m,i} > \theta_m \\ 0, & \text{otherwise} \end{cases}$$

$$F_{m,i} = \begin{cases} \alpha_{m,i}, & \text{if } f_{m,i} > t_{m,i} \\ \beta_{m,i}, & \text{otherwise} \end{cases}$$

$f_{m,i}$ is the weighted sum of the 2-D integrals. is the decision threshold for the *i*-th feature extractor. $\alpha_{m,i}$ and $\beta_{m,i}$ are constant values associated with the *i*-th feature extractor. θ_m is the decision threshold for the *m*-th classifier.

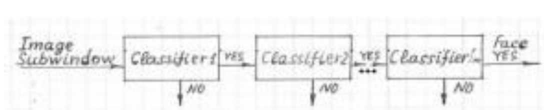


Fig.3 Object detection Viola-Jones filter

ISSN:-2581-6934|www.jidps.com

The cascade architecture is very efficient because the classifiers with the fewest features are placed at the beginning of the cascade, minimizing the total required computation. The most popular algorithm for features training is AdaBoost.

SYSTEM IMPLEMENTATION

First off, Face detection and Face recognition are two totally different things although one builds upon the other (recognition builds upon detection). Detection is the process by which the system identifies human faces in digital images, regardless of the source while Recognition is the identifying a known face with a known name in digital images, still regardless of the source. The source can range from a scanned copy of a photograph to a live video stream. Face detection and recognition is a section of Machine learning with a good number of research topics focused on improving the existing algorithms.

What is EmguCV?

Emgu CV is a cross platform .Net wrapper to the OpenCV image processing library. Allowing OpenCV functions to be called from .NET compatible languages such as C#, VB, VC++ etc. The wrapper can be compiled by Visual Studio, Xamarin Studio and Unity, it can run on Windows, Linux, Mac OS X, iOS, Android and Windows Phone.

the process of detection and recognition into the following steps:

- Getting the Camera to work i.e to provide live feed
- Detecting human faces from the feed provided by the camera
- Saving the detected face with an identifier
- Training our recognizer with the saved faces
- Carrying out recognition against the trained system

Classifier/detector's DetectMultiScale method takes four(4) parameters. The parameters are:The first parameter is the grayscale image i.e the actual image we want to detect the face from. Notice it has been converted to a gray frame using the Convert in the code snippet above.

The second parameter is the scale factor. This parameter must be greater than 1.0 and the closer it is to 1.0 the longer it will take to detect faces but there's a greater chance that you will find all the faces. I don't mind my app taking its time to give me adequate result.The third parameter is the minimum number of nearest neighbors. The higher this number the fewer false positives you will get.The last parameter is the max size in pixels. Notice i set to empty.Please keep in mind that the other three (3) parameters are optional, while the only required parameter for the method is the Image.

That's about it for face detection. In the second part of this article, i would discuss and show how face recognition can be implemented. How you can train your recognition engine and store training data, and how based on the training data, you can accurately predict (or at least try to) which face is supplied.

Implementing a Working Camera

To get live feed from our installed camera, we would need to import EmguCV UI tools into our visual studio. To do that Add ImageBox Control.
Drag the control onto the form and rename it to imgCamUser, you can re-size as you please.
After that has been added successfully, we use the Capture class to get the live feed from our Camera.

Implementing Face Detection

live feed of camera need to be able to detect face(s) from the feed.Popularly used for face detection is the Viola-Jones algorithm because it is the easiest ready to use face detection method which is supported by EmguCV and has proven to return great results. In using this algorithm, we would be making use of a face detector called Cascade Classifier/Detector that has been trained on thousands and thousands of human faces (remember face detection is a subject under machine learning). The training data generated from the faces are stored in xml files which come by default with the EmguCV package which are stored in [EmguCVRootFolder]\opencv\data\haarcascade.

The method can be attached to a user triggered event e.g. After a face is detected, the user clicks a save button that calls this method. It can also be automatically called when a face detection happens. The code above also displays a dialog box

ISSN:-2581-6934|www.jidps.com

prompting the user to supply a username for the face that is about to be saved.

To make the training and recognition engine work better, we need to capture multiple faces of the same user. Our username is not a key in our database table, so we can have multiple records with the same username. Having multiple faces of a user helps the engine get trained better. Face recognition is field in Machine learning machine learning greatly relies on a lot of existing data in order to make accurate/near accurate predictions in the future.

Training the recognition engine

After having saved our data, we need to train our recognition engine with the dataset are available. The time at which this takes to complete a training sequence is determined by how large your dataset is. Below shows the code that trains our recognizer engine. a new instance of EigenFaceRecognizer which is an EmguCV class as _faceRecognizer, then we proceeded to call all the faces we have stored in our database as List< Face >. Each of the faces were converted to Image < Gray, byte > and resized and passed to the Train method of the EigenFaceRecognizer class. The Train method takes two main parameters; the first is the Array of Images, and the second is the Array of Integer Labels of the images. The label is what is returned when the face is recognized, which we would see how that is done later. After training our engine, we need to save the current state of the model, in order to carry out recognition. EmguCV allows the model state to be saved as a YAML file. You can carry out more training as soon as you have more faces in your data store.

Recognizing Faces

So far, we have been able to store our training data (detected faces), trained our recognizer engine, store our engine model state. The Predict method of the recognizer engine to recognize the specified face. The method takes just one parameter which is the Image of the face we are trying to recognize or predict. 0 is returned if the face cannot be predicted, and the Integer Label is returned if it was predicted successfully.

Conclusion and Future Enhancement

Conclusion

We have presented automatic techniques for face segmentation and landmark detection in frontally posed range images using only depth information as input. Our segmentation approach extracts the entire face region by combining edge detection, region clustering, and shape analysis. Our landmark detection approach uses surface curvature classification and relief curves of face images to detect the nose tip, nose corners, and inner eye corners.

The entire face region was correctly extracted by our approach in 99.3%, 99.7%, and 96.1% of the images of the FRGC v1.0, FRGC v2.0, and BU-3DFE databases, respectively. There was an increase in the verification rate of an ICP-based recognition system of 2% at 10^{-3} FAR in comparison to the two other segmentation approaches and also an increase in the verification rate of 0.4% at 10^{-3} FAR when using our face recognition approach.

The desired landmarks were correctly found in 99% of the FRGC databases and also 99% for nose landmarks and 93 % for eye corners in the BU-3DFE database. The nose tip was the most precise landmark, presenting the lowest mean error and the highest detection rates for all databases. Our landmark detection approach obtained equivalent or better performance when compared to three other techniques and is also robust to facial expressions, which can be seen in our experimental results for FRGC v2.0 and BU-3DFE databases. The detected landmarks were also employed to extract different rigid regions of the face. These rigid regions were employed in a recognition experiment in order to evaluate the improvement obtained in verification rates when face images containing expression variations were used as probe images. Verification rates up to 46% higher were obtained for images with high levels of expressions, 36% higher for images with normal expressions, and 5% higher for images with neutral expression.

Finally, our results show the benefits of both face detection and landmark detection in a face recognition system and how much they improve its verification rates. Our presented approaches were encapsulated in our automatic 3-D face recognition system based on SA + SIM, and we obtained a

ISSN:-2581-6934|www.jidps.com

verification rate of 99.9% at zero FAR for neutral images and 96.5% at 10^{-3} FAR when all images of the FRGC v2.0 database were matched against each other (i.e., including matches between images with non- neutral expressions).

Future Enhancement

Enhancement in the methodology of face recognition and detection is identifying multiple personalities in a single image. Robustness in the natural environment of landmark detection. Enriching the recognition technique further more by concentrating more on recognition even if the illumination of the image changes rapidly.

References:

1. A. Colombo, C. Cusano, and R. Schettini, 2006, “3D Face detection using curvature analysis”, Pattern recognition Volume 13, no. 3

2. A.B. Moreno, A. Sanchez, J. F. Velez and F. J. Diaz, 2003, “Face recognition
using 3D surface – extracted descriptor in procirish mach.

3. A. M. Bronstein, M. M. Bronstein and R. Kimel, August 2005, “3 Dimensional face recognition” int. J. Comput Vol. 64.

4. C. Hesher, A. Srivastava and G. Ealevacher, 2002 “Principal component analysis of range images for facial recognition in proc. Into. Conf. imaging sci.

5. C.C. Queirolo, L. Silva, O.R.P. Bellon and M. Pamplona Segundo Feb 2010, “3D face recognition using simulated annealing and surface interpenetration measure” IEEE trans. Pattern Analmach.intel, Vol. 32.

6. D. Colbry, G. Stockman and A. K. Jain 2005 “Detection anchor points for 3D verification” In proc. IEEE comput.soc.conf.comput.vis.patternrecog.

7. F. Tsalakanidou, S. Malassiotis and M. G. Strintzis Feb 2005, “Face localization and authentication using colour and depth images” IEEE trans pattern Anal machintel Vol. 14.

8. G. Pan, Y. Wu and Z. Wu 2003, “Investigating profile extracted from range data for 3D face recognition in proc IEEE int.conf.syst.,man,cybern.

9. I. Kakadiaris, G. Passalis, G. Toderici, M. Murtuza, Y. Lu, N. Karampatziakis and T. Theoharis April 2007, “3D face recognition in the presence of facial expression : and annotated deformable model approach”. IEEE trans pattern anal.mach.intel Vol.29.

10. J. Y. Cartoux, J. T. Lapreste and M. richetin 1989, “Face authentication or recognition by profile extraction from range images”. In proc.IEEEcomput.soc.work interpenetration 3Dimensionl scenes.

ISSN:-2581-6934|www.jidps.com