International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 04 | Apr 2020 www.irjet.net p-ISSN: 2395-0072

Emotionalizer: Face Emotion Detection System

Sweety Malviya[1], Mayuri Rathore[2], Vanshika Parihar[3], Twinkle Rathore[4], Shreyansh Malvi[5], Sanjay Kalamdhad[6]

[1-7]*Dept. of Computer Science and Engineering, Shri Balaji Institute of Technology and Management, Betul, M.P*

---***---

Abstract - *This project is mainly focused on emotion recognition by face detection. Facial expression is key aspect of social interactions in various situations. We used synthetic happy, sad, angry, fearful, disgust faces determining the amount of geometric change required to recognize these emotions. Emotion is a part of a person's character that consists of their feelings as opposed to their thought that is the key point of emotionalizing and analyzing each and every emotion by software i.e. able to read emotions as well as our brains do. It is basically developed on python using machine learning.*

Key Words: Facial emotion detection, facial recognition, facial analysis.

1. INTRODUCTION

Emotionalizer as the name combined from the word emotion analyzer and the project as well analyzes human emotion by matching them to specific features regarding that particular expressions. Face recognition is an important part of the capability of the human perception system and is a routine task for humans while building a similar computational model of face recognition. The computational model not only contributes to theoretical insights but also to many practical applications like automated crowd surveillance, access control, design of human-computer interface (HCI), content-based image database management, criminal identification and so on. The earliest work on face recognition can be traced back at least to the 1950s in psychology and to the 1960s in the engineering literature. Some of the earliest studies include work on facial expression emotions by Darwin. But research on automatic machine recognition of faces started in the 1970s and after the seminal work of Kanade. In 1995, a review paper gave a thorough survey of face recognition technology at that time. At that time, video-based face recognition was still in a nascent stage. During the past decades, face recognition has received increased attention and has advanced technology. Many commercial systems for still face recognition are now available. Recently, significant research efforts have been focused on video-based face modeling/tracking, recognition and system integration. New databases have been created and evaluations of recognition techniques using these databases have been carried out. Over the last few decades, lots of work is been done in face detection and recognition. A facial recognition system is a computer application capable of identifying or verifying a person from a digital image or a video frame from a video source. One of the ways to do this is by comparing selected facial features from the image and a facial database.

Fig 1- Configuration of a face recognition system [1]

Face detection is a computer technology being used in a variety of applications that identifies human faces in digital images. Face detection also refers to the psychological process by which humans locate and attend to faces in a visual scene. It is typically used in security systems and can be compared to other biometrics such as fingerprint or eye iris recognition systems. Recently, it has also become popular as a commercial identification and marketing tool. Face detection can be regarded as a specific case of object-class detection. In object-class detection, the task is to find the locations and sizes of all objects in an image that belongs to a given class. Examples include upper torsos, pedestrians, and cars. Face-detection algorithms focus on the detection of frontal human faces. It is analogous to image detection in which the image of a person is matched bit by bit. Image matches with the image stores in the database. Any facial feature changes in the database will invalidate the matching process. A reliable face-detection approach based on the genetic algorithm and the eigenfaces technique.

2. PROBLEM DEFINITION

In the early few years, several papers have been published on face detection in the community which discusses different techniques like a neural network, edge detectors and many more. There is a good survey by Chellapa, Wilson, and Sirohey (1995) which tells about the trends of paper in face detection. Previously, many researchers and engineers have designed different purpose-specific and application-specific detectors. The main goal of this kind of classifiers was to achieve a very high detection rate along with a low computational cost. Few examples of different detectors are corner detectors, AdaBoost Haar Cascade detector by Jones and Viola [10] are very useful. This kind of detector mostly uses simple and fast classifiers that reject the most common negative samples and then they use progressively more complex classifiers to deal with the more difficult and odd negative samples. Another approach to detecting the face is through a skin color classification algorithm [4]. Kjeldsen and Kender used the concept of different color space model to separate the skin patches. But if those algorithms are used solely, then it becomes difficult to detect more than one face from the image. Hsu, Abdel-Mottaleb, and Jain came up with the tactics of calculating the eye, mouth and nose map values [4]. Eyes, nose, and mouth are significant features of the face that distinguishes face from other 12 parts of the body. Studies have shown that they exhibit unique properties in the Grey color model, so it becomes easy to detect the face region using this feature extraction. Poggio, Heisele, and Ho presented a component-based approach of locating facial components, extracting them and combining them into a single feature vector which is used for classification of faces by Support Vector Machine. The authors used the grayscale image to define the feature vector for classification. Authors' using the approach of Support Vector Machine iterates through the whole image and compares it with a face template to classify the region of interest. This takes a very high computation time and error rate. Also, some of them can detect faces only from grayscale images. But our algorithm overcomes these drawbacks of classification. In our approach, preprocessing stages are applied which extracts the region of interest and feature vector is applied to this ROI instead of the whole image for classification of the face region. Detecting the face from color images poses various difficulties under varying lighting conditions, pose change and when there are additives on the face region like beard, mustache, etc. To overcome this, we applied certain preprocessing stages to my algorithm so that it detects the face region accurately with less error rate and low computational cost. Irjet Template sample paragraph. Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, sc, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

3. OBJECTIVE

In the first step, we would haul out and identify the face region based on the skin color segmentation algorithm. There are lots of variations in the human race or lighting condition, so to accurately detect the skin patch region we converted the image into RGB, Grey and HSV color space. This would take advantage of all models and find out all the skin regions from images. Afterward the face feature 13 extraction process is applied which calculates the map values of different face features[4]. These threshold values and preprocessing values are passed to the feature vector of Linear SVM which would classify the image in the face and non-face class. This classification is based on the training of the data set. Our data set consists of around 125 images with 305 faces. The images are taken from the internet and from my collection of photographs. The next section discusses computer vision concepts used in this algorithm and approaches adopted by other authors. Section 3 describes the detailed approach and explanation of our algorithm. It gives step by step elucidation of how each step is executed in our algorithm. Section 4 discusses the results obtained from our algorithm and shows the comparison with the Haar cascade classifier method. Section 5 points out the limitations of our algorithm and displays a few scenarios on which our algorithm does not show satisfactory results. In section 6 we listed out the future work which we will adopt in the current methodology. Section 7 discusses the final conclusion and findings from our approach.

4. LITERATURE SURVEY

In this section, we discuss various concepts like skin color detection, edge detection, morphological operators and support vector machine used by prominent authors of face detection.

4.1 Skin Color Detection

Many algorithms have been developed to determine the face using the skin color. There have been past findings to detect the skin color by determining the geometric correlation between the position of hair and face. In that, the authors used the normalized RGB model to extract skin.

r - (eq . 1)
g - (eq . 2)

For determining the skin color from the input images we have to convert the image into different color spaces like RGB and HLS. Afterward they process each image to pull out the skin color utilizing the properties of color space. The implementation of this is very simple and depends upon the threshold value of the skin color. The first step is to mask the skin color and then extract the part of the image which resembles the skin. The skin color detection has the limitation of detecting only one face per image.

4.2 Edge Detection

In the edge detection phase, edges of the object from the images are detected using a particular threshold value. Using the Gaussian filter the noise from the images is filtered out. The next step would involve calculating the intensity gradient of the image. This would help eliminate the varying lighting condition. Then the morphological operators like elation and dilation are applied which reduces the noise from the background image. The non-suppression is applied which helps in removing the points that are not part of the edge. The most important step of this phase is to detect the edge of the objects. So, dependingupon thethresholdpixeltheedgesareacceptedorrejected.Thevalueof the threshold is selected based on the average of the tested training data set of images.

4.3 Extracting feature of face

Eyes, nose, and mouth are the most prominent facial features which help in detecting the face. Based on the luminance and chrominance values we can locate the boundaries for eyes, nose, and mouth from the image [2]. Luminance (Y) represents the brightness in the image i.e. the black and white portion of the image. It represents the details of the image without any of its color information whereas the chrominance represents the color information. Chrominance is represented as two color difference: where Cb is defined as Blue-Y' and Cr is defined as Red-Y'.

Eye Map - In this, we calculate the luminance component and chrominance component of the eyes. The results of both these are added and combined. These values are calculated based on the properties of the higher value of chrominance and lower value of luminance found near eyes [2].

Mouth Map - The mouth color has a higher red component and lowers blue component as compared to supplemental regions of the face. So, we can conclude that the chrominance is greater than the luminance component.

4.4 Color Space – RGB, HSV

Active development in the content-based image field has led to great concentration to the study of skin color classification. Locating different image objects like face, eye, car, etc. can be exploited for different purposes like recognition, editing, detection, indexing, and various other interactive purposes. Tracking the position of the face using skin color also provides a vital stepping stone in studies related to facial expression. Recent researches in the algorithms for face localization mostly prefer to utilize the color information to approximate the skin color. Skin color classification can be considered as an important task as most of the prevalent algorithm for face identification use color information to estimate skin region. Estimating the skin color region is most often considered as the vital step of the face localization process. Nowadays, most of the research development in face detection using skin color is established on the concept of HSV, RGB and Grey color spaces. In this section, we describe brief information about these different color models.

RGB

The perceived human color is proportional to the varying condition of illumination. Using the normalized color histogram we can sense the pixels for skin region which can be further normalized to resist the varying luminance condition. The Red, Green and Blue vector of the image is converted into a normalized form which provides a rapid means for detecting the skin and thereby confines the face region. There are limitations to this algorithm if there are some more skin regions like hands, neck, legs, etc. in the image then it cannot detect it properly.

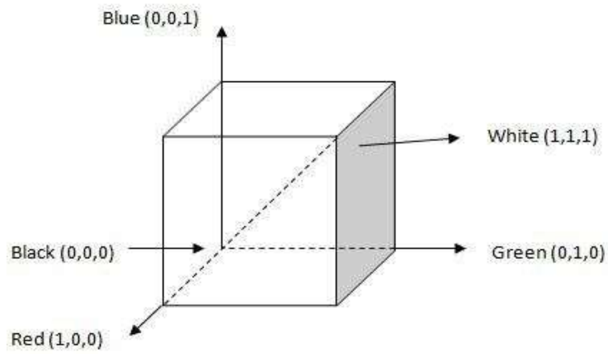

Fig -2: Color Cube Image

HSV

We make use of the HSV space to separate the skin regions from the background. Classification in HSV color space is the same as Grey color space but here the accountable values are saturation (S) and hue (H) instead of luminance and chrominance. A pixel is categorized to have a skin tone if the value of Hue and Saturation fall within the specified threshold value and the distribution gives the localized face image.

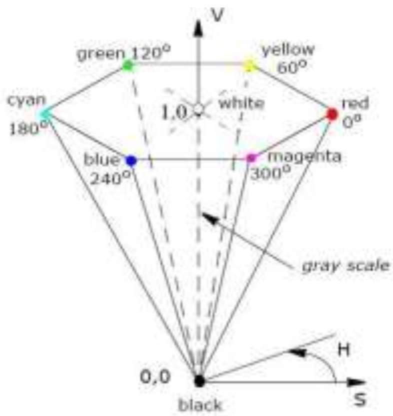

Fig -3: HSV Color Model

4.4 Support Vector Machine

In machine learning, the task of deducing a category from supervised training data is known as Supervised Learning. In supervised learning, the training data consist of a set of training examples, where each example is a pair consisting of an input and an anticipated output value. A supervised learning algorithm analyzes the training data and then predicts the correct output categorization for given data-set input. For e.g. Teacher teaches the student to identify apple and oranges by giving some features of that. Next time when a student sees apple or orange he can easily classify the object based on his learning from his teacher, this is called supervised learning. He can identify the object only if it is apple or orange, but if the given object was grapes the student cannot identify it.

Support Vector Machine iterates through the whole image and compares it with the face template to classify the region of interest. This takes very high computation time and error rate. Also, some of them can detect faces only from grayscale images. But our algorithm overcomes these drawbacks of classification. In our approach, preprocessing stages are applied which extracts the region of interest and feature vector is applied to this ROI instead of the whole image for classification of the face region. Detecting the face from color images poses various difficulties under varying lighting conditions, pose change and when there are additives on the face region like beard, mustache, etc. To overcome this, we applied certain preprocessing stages to my algorithm so that it detects the face region accurately with less error rate and low computational cost.

Using SVM for Face Detection

The algorithm proposed by Osuna, Freund, and Girosi detects faces by exhaustively scanning an image for face-like patterns at many possible scales, by dividing the original image into overlapping sub-images and classifying them using an SVM to determine the appropriate class [3]. Multiple scales are handled by examining windows taken from scaled versions of the original. Before storing the image some pre-processing steps like masking, illumination and histogram equalization are performed. In the masking process, unnecessary noise like the background pattern is reduced from the objects of interest. And then histogram equalization is used that manages the distribution of colors in images.

The images of a class face and class non-face are used to train the SVM using the kernel and upper bound margin values. Once a decision surface has been obtained through training, the run-time system is used over images that do not contain faces, and misclassifications are stored so they can be used as negative examples in subsequent training phases. In order to increase the precision of detecting face, we can use negative examples for training misclassification class. There are ample non-face images available that can be trained in SVM. Non-face images are richer and broader than face images.

4.5 Haar Cascade Classifier

In statistical model-based training, we take multiple positive and negative samples and extract different features from these samples. These distinctive features are then compressed into statistical model parameters which are used as a special property to classify different objects. By making adjustments in these parameters we can improve the Accuracy of classification for these algorithms.

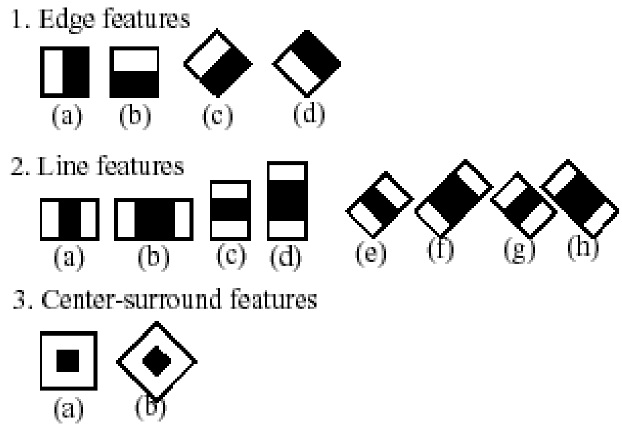

Fig -4: Haar Cascade Classifier

The fundamental concept for detecting objects from images for the Haar classifier is the Haar-like features. These features exploit the difference in contrast values between contiguously grouped pixels instead of using the intensity values of that particular pixel. These contrasting values between the grouped pixels are used to detect relative light and dark spots from the images. These two to three contiguous groups with a comparatively contrasting value form a Haar-like feature. In images we have the object of different sizes, these Haar features can be scaled by increasing or decreasing the size of the grouped pixel being examined. This scaling of the pixels makes it possible to detect and extract objects with varying sizes.

4.6 Programming in OpenCV

OpenCV acronym for Open Source Computer Vision Library is a library containing functions for computer vision. It is developed by Intel and now handled and supported by Willow Garage. The library is a functional cross-platform and runs on Windows, Android, FreeBSD, Maemo, iOS, OpenBSD, Linux, and Mac OS. The current release of the library is obtained from the Sourceforge and they also provide the binaries for the user, so that they can develop according to their requirements. OpenCV makes use of CMake to compile source files to start using the library. The main focus of this library is on the real-time image processing functionality and implementing the machine learning algorithms. By using this we can improve the cost of computation and take an initiative to advance the CPU – intensive applications. The areas of application where OpenCV can be useful are facial recognition system, mobile robotics, gesture recognition, segmentation, object identification, motion tracking and many more. OpenCV also includes a statistical machine learning library that supports the above areas of application. The name of the functions that support this library is decision tree learning, expectation-maximization, gradient boosting trees, Naïve Bayes classifier, k- nearest neighbor, artificial neural network, support vector machine(SVM) and many more.

4.7 Tool and Image Dataset

The tools which I have used in my projects are Visual Studio code and Git. The image dataset used in this project was developed from the images which I had clicked from my camera. Also, some photos were taken from the internet (not containing copyright issues) to test varying conditions.

5. ALGORITHM

In this section, we will discuss our algorithm which we have developed using different concepts as explained earlier. Convert the input image into a different color space models with the goal of obtaining a specific region from the image as RGB space.

5.1 Convert Image into RGB and HSV color model and obtain Binary Image.

RGB Model

Red, Green and blue are the three additive primary colors for the RGB color space. A combination of these components in assorted proportions would produce different colors. The RGB model is represented by a three-dimensional cube with red, green and blue defining each axis. Black is defined at the origin (0, 0, 0) whereas white is at the other side the cube i.e. at (1, 1, 1). The gray color follows the diagonal line alongside the color cube starting from white to black. The image consists of three channels 8 bits each, with each color representing different channels. The RGB color model condenses the design of computer graphics systems but is not ideal for all applications as all three components are highly correlated with each other, making it difficult to implement different image processing algorithms as they will be not resistant against different lighting conditions. Here we analyze the image for skin region, so we will look for a combination of R, G and B components in such a way that resultant color is the skin color.

HSV Model

Skin color extraction in the HSV model is similar to that of the RGB model, but the accountable values here are Hue and Saturation. The HSV color space helps to distinguish the skin regions from the background. Analogous to the model the Hue values are chosen between h_min to h_max whereas the saturation values are taken from s_min to s_max. Depending upon this range, a pixel is classified as a skin tone.

5.2 Face Feature Extraction

Next preprocessing stage involves extracting different face features like eyes, mouth ,and nose from the detected blobs. Once the blobs are detected the extracted face region becomes the Region of Interest (ROI), and is converted into YCbCr model space. This image is then split into three different channels of chrominance and luminance. Then using eq. (3), eq. (4) and eq. (5) we calculate the eye map values and mouth map values. Then we will combine these images and apply the filtering operation of dilation and erosion [4]. This will yield the eye and mouth region.

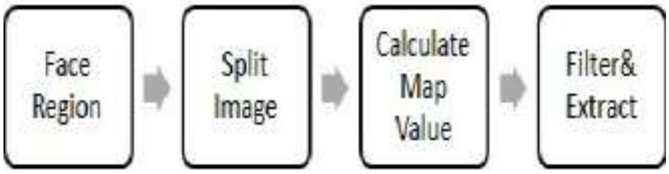

Fig -5: Face Feature Extraction

After locating the eyes and mouth region the geometrical position of the spots is checked. If the eyes and mouth center forms a structure that is equivalent to the equilateral triangle, then we can confirm the position of the face. The locations of the center of these regions are passed over to the next stage for classification, which checks over with face template from the data set. Dataset will gather the data regarding any specific expression and then by the help of the dataset it will provide emotion analysis regarding that specific features.

5.3 Classification of ROI using SVM

There is a large difference between the face and other parts of the body containing the skin. Features like rich texture, eyes, mouth create a huge difference between the face and other body parts like hands, arms, shoulders, legs, and neck. The Linear SVM kernel function $K(x_i , x_j) = \langle x_i , x_j \rangle$ is designed to classify the data into face and non-face and we used the OpenCV library to train the samples. As the number and size of faces are often different in images, in order to detect all the faces, we have to take pyramid analysis. In that case, we scale every image a few times until the size of the skin color region is fixed. In every scale, we scan the original color image from left to right, top to bottom in the effective skin color regions, and intercept image as a detected sample. After the detection sample is processed with a mask, the sample feature vector is put into the SVM classifiers to classify. This classifier uses the data from the pre-processing stages.

5.4 Final Result

We can observe the final result of the image. All the faces in the input Here image are detected. The results and limitations of the algorithm are discussed in the next section. After the SVM classifies the region into a face label, we highlight that part of the original image bounded with a rectangle of red color after capturing a still image. Analyze the expression regarding to the face and then it will provide the outcome among all the expressions as the human is sad, happy, disgust, fearful, angry and the outcome could also be changed if required, as sometimes it can cause some mismatch regarding to specific feature and then it can be updated on the system so the accuracy of the result becomes higher.

6. RESULT AND LIMITATION

Result

The proposed algorithm was trained and evaluated on the dataset of around 125 images containing 305 face images. This dataset was build from my collection of photographs and some random images from the internet. The test images consisted of images with different lumination conditions – night time, daytime and combination of them. The image formats acceptable to the algorithm are jpeg, png, bmp, etc. The dataset consists of images of size ranging from 400x320 to 2000x1800. If the size of the image is more than 2000x1800 then it would create a problem in processing the image. We implemented the algorithm on an Intel® Core™ i5 CPU M 430 @ 2.27 GHz with 4.00 GB of memory.

Limitation

There are certain limitations to my proposed algorithm for face detection. First of all, I have carefully analyzed my data set and based on that I have kept the threshold ranges for detecting skin region. If due to varying conditions, the skin color does not fall into a specified range than my preprocessing stage will not be able to detect the skin region. Sufficient cautionary steps are taken to detect the skin region by using three different color models, but there may be times when skin region is not detected. Second, if the face in the image is tilted by some angle then it cannot detect the face because we consider the image height and width during certain calculations like merging, overlapping. If the face is tilted then the height and width are changed, so the supposed face region is automatically neglected in the pre-processing stage. Also, my algorithm calculates map values during the face feature extraction process. This is highly dependent on the visibility of face features. For example in the below figure the person in the middle has covered his eyes, in that case, the face is not detected.

7. FUTURE WORK

In-depth research for detecting the variation in human pose should be carried out. The current algorithm needs some efforts in detecting various poses from images. Also, experiments should be carried out to observe and analyze different kernels for classification using Support Vector Machine.

8. CONCLUSION

In this project, I have presented face detection algorithm using the skin color detection, edge detection, facial feature extraction and using the concept of different color space. After these pre-processing stages, the algorithm utilizes the highly powerful concept of Support Vector Machine (SVM) to classify the image into the face and non-face region. We have significantly reduced the misclassification errors as compared to the Adaboost classifier of Viola and Jones [10]. The computation time for our algorithm is very less and the accuracy on the image data set of 125 images with 305 face image is around 90% with an error rate of approximately 16%. We overcame the limitation of detecting one face from the image using a skin color algorithm; by combining the concept of different color space and face feature extraction process.

9. REFERENCES

[1] Garcia, C., & Tziritas, G. (1999). Face detection using quantized skin color regions merging and wavelet packetanalysis.Multimedia,IEEETransactionson,1(3), 264-277.

[2] Hsu, R. L., Abdel-Mottaleb, M., & Jain, A. K. (2002). Face detection in color images.Pattern Analysis and Machine Intelligence, IEEE Transactions on, 24(5), 696-706.

[3] Osuna, E., Freund, R., & Girosi, F. (1997). Training support vector machines: An application to face detection. Paper presented at the Cvpr, 130.

[4] Rein-Lien Hsu, Abdel-Mottaleb, M., & Jain, A. K. (2002). Face detection in color images. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 24(5), 696-706.

[5] Rowley, H. A., Baluja, S., & Kanade, T. (1998). Neural network-based face detection. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 20(1), 23-38.

[6] VIOLA, P. AND JONES, M.J. 2004. Robust real-time face detection. International journal of computer vision 57, 137-154. .

[7] Yang,G.,&Huang,T.S.(1994).Humanfacedetectionina complexbackground. Pattern Recognition, 27(1), 53-63.

[8] Shih, P., & Liu, C. (2006). Face detection using discriminating feature analysis and support vector machine.

[9] KADIR, T., BRADY, M. 2001. Saliency, Scale and Image Description. International Journal of Computer Vision, Volume 45, Number 2.

[10] P. Viola and M. Jones.(2001). Rapid Object Detection UsingaBoostedCascadeof Simple Features. IEEE Conf. CVPR, Vol. 1, pp. 511-518.