IOSR Journal of Engineering (IOSRJEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
Special Issue || Dec. 2019 || PP 61-65
International Conference on Management Practices, Innovations & Research 2019

Real Time Filtering Using Convolutional Neural Network

Shriya Sharma[1], Mona Bhandekar[2], Kanchan Ingole[3], Pratiksha Khedkar[4], Ayushi Umredkar[5]

UG Student, Dept. of Studies in CSE, TGPCET College of Engineering, Nagpur, Maharashtra, India1 Assistant Professor, GulrukhNazneenDept. of CSE,

Abstract: *In this project, we aim to Implement a real-time expression recognition and tracking the head poses position from high definition video using Haar Classifier. SimpleCv and OpenCv libraries are used for face recognition and tracking the head poses position. Face detection and recognition from an image or a video is a popular topic in biometrics research. Face recognition technology has broadly attracted interest due to its massive application value and market potential, such as real-time video surveillance system. It is broadly acknowledge that the face recognition has played an important role in surveillance system as it doesn't need the object's co-operation. We propose a real-time face detection system based on IP camera and image set algorithm by way of OpenCV and Python programming development. The projected framework in this project has the main objective of classifying the facial expression shown by a person. These classifiable expressions can be any one of the six general emotions. After the initial facial localization is performed, facial landmark detection and feature extraction are applied where in the landmarks are determined to be the fiducial features: the eyebrows, eyes, nose and lips. This is mainly done using the Sobel operator and the Hough transform followed by Shi Tomasi corner point detection. This leads to input feature vectors being formulated using Euclidean distances and trained into a Multi-Layer Perceptron (MLP) neural network in order to classify the expression being display. The results achieved have further deal with higher equivalence in certain emotions and the essentially subjective nature of expression.*

Facial key points include points around the eyes, nose, and mouth on any face and are used in many applications, from facial tracking to emotion detection. The partly complete module should be able to take in any image containing faces and identify the location of each face and their facial key points. The projected facial recognition system uses few of the many computer vision algorithms built into the OpenCV library and are implemented at the basic level. This extensive computer vision library is open source and is still growing. The proposed system does real time filtering and facial key point detection. This implementation uses a Convolution Neural Network to train the system at each step, visualize the loss and learn in the next detection.

Keywords: *Python, Android studio, Anaconda python, MATLAB, JAVA, Javascript.*

I. Introduction

This project aims to explore the world of camera calibration to find known facial landmarks, to calibrate the orientation of the face, to generate a pose, and to apply various effects to the user's facial structure spanning flat portable network graphics distortion and applications. In addition to this, the project also aims to tackle the filter known as a face swap using the Orthogonal Procrutes Problem as a method to handle and mitigate orthogonal calibration issues for swapping facial features with another image of a differing projection.

To understand how this augmenting of reality works, the process of generating a known calibration in 3D from a 2D image representation of the world must be traversed. Using various methods of detecting the locations and classifications of features, a temporal perspective axis can be generated and used to augment and distort these effects to the desired location on the user's face.

Face recognition and land marking are often used as the backbone to many camera based application. Features, such as the auto-focus build into most recent day cameras use a face detector to recognize the regions that involve focusing. Commercial applications, such as Snapchat, BeautyPlus are based around recognizing the facial region and performing image manipulation to the facial features, these applications are extremely popular and have a high number of users world-wide. Many mobile applications rely on the use of machine learning algorithms to identify patterns in image data. Machine learning uses pre-annotated data to learn the difference between the data values, but it requires a large amount of annotated data to learn which is time consuming even with the available tools. However, Mathias showed thatif a strict data annotation regime was implemented,a small dataset can match the accuracy of methods trained on large datasets. We aim to analyse how well neural networks can be integrated into mobile devices to increase the accuracy and reliability of the applications.

Many modern mobile applications include face recognition and landmarking into their systems, such as Snapchat, beauty filters and camera auto-focusing systems, where they apply regression based machine learning algorithms for correct face landmark detection, allowing the manipulation of facial appearance. The mobile applications that include machine learning have to conquer issues such as lighting, occlusion, camera quality and false detections. A solution could be provided through the resurgence of deep learningwith neural networks, as they are showing significant improvements in accuracy and reliability in comparison to the state-of-the-art machine learning. Here, we express the process by using trained networks on mobile devices and analysis its effectiveness. We also compare the effect of employing max-pooling layers, as an efficient method to decrease the required processing power. We compared network with 3 different amounts of max-pooling layer and ported one to the mobile device, the other two could not be ported due to memory restrictions. We will release all code to build, guide and use the model in a mobile application. The results show that in spite of the limited processing capability of mobile devices, neural networks can be used for difficult challenges while still working in real-time. We show a network running on a mobile device on a live data flow and give a suggestion on the structure of the network.

II. Background

Deep learning and facial landmarking are well researched areas. Deep learning can perform many different tasks, such as object classification and segmentation, to do these the networks structure and layer methodology must be suited to that task. Facial landmarking is best suit for a regression based approach as it has been commonly proven most effective, because the landmarks will shift with different facial expressions.

Lopes, Andre Teixeira, Edilson de Aguiar and ThiagoOliveiraSantos “A Facial Expression Recognition System Using Convolutional Networks”,IEEE(2015). [1] For facial landmarking ,the data should be preserved as much as possible , as it is easy to lose key facial features, such as a eye corners and nose tip when employing max-pooling layers.

Oquab, M.Bottou, Laptev and I.Sivic , CVPR(2014) [2]“J Learning and Transferring mid-level image representation using convolutional neural networks” . A simple transfer learning procedure yields state-of-the-art results on challenging benchmark datasets of much smaller size.

Kazemi, V. & Sullivan, J, 2014 ,[3] “One Millisecond Face Alignment with an Ensemble of Regression Trees”. An ensemble of regression trees can be used to regress the location of facial landmarks from a sparse subset of intensity values extracted from an input image. The presented framework is faster in reducing the error compared to previous work and can also handle partial or uncertain labels.

Abhishek G C1, Pramukh B V1, Pranav T V1, Shravan S Vasista1, B S Prathibha (2018) , [4],

“Facial Key-Point Detection and Real-Time Filtering Using Convolutional Neural Network”.Pre-trained HaarCascade[1] Classifiers are used to detect faces in an image and also to recognise the position of the eye in the detected face. The Network architectures proposed in this paper effectively reduces processing time, improves accuracy of the CNN running on low-end GPUs or even on CPUs. The OpenCV library which is used for image processing is an expanding library and hence this system is built to incorporate any changes in the library.

III. Existingsystem

In the existing system, Face recognition and land marking are often used as the backbone to many camera based application. Features, such as the auto-focus build into most recent day cameras use a face detector to recognize the regions that involve focusing.

Commercial applications, such as Snapchat, BeautyPlus are based around recognizing the facial region and performing image manipulation to the facial features, these applications are extremely popular and have a high number of users world-wide.

Many mobile applications rely on the use of machine learning algorithms to identify patterns in image data. Machine learning uses pre-annotated data to learn the difference between the data values, but it requires a large amount of annotated data to learn which is time consuming even with the available tools. However, Mathias showed that if a strict data annotation regime was implemented, a small dataset can match the accuracy of methods trained on large datasets. We aim to analyze how well neural networks can be integrated into mobile devices to increase the accuracy and reliability of the applications.

Many modern mobile applications include face recognition and land marking into their systems, such as Snapchat, beauty filters and camera auto-focusing systems, where they apply regression based machine learning algorithms for correct face landmark detection, allowing the manipulation of facial appearance. The mobile applications that include machine learning have to conquer issues such as lighting, occlusion, camera quality and false detections.

Snapchat filters area unit fascinating progressions inside the scope of increased reality. The developments that are applied to the scope of standardization and augmenting of real and faux along into one application within the palm of one’s hand has allowed for fascinating deviation of reality to exist.

As Snapchat grows in quality, the standard and variations of the augmenting of what the lens captures expands perpetually. during this project, I explored a set of the speculation concerned in making such a filter – spanning over estimating create through landmark machine learning, augmenting easy flat objects to the create of the face, and exploring the Orthogonal Procrustes drawback thoroughly.

IV. Methodology

This facial-keypoint recognition system can be built and run on any platform as it uses an open source Keras module with Tensorflow backend and OpenCV library. The modules can be programmed using Python. The system comprises of three module.

4.1 Pre-Processing and Face Detection

This module is aimed towards investigating the OpenCV library to process images from a dataset and apply the tool to detect faces in the images.

4.1.1 Detect Faces Using a Haar Cascade Classifier

At its root face detection is a classification problem - that is a problem of distinguishing between distinct classes of things. With face detection these distinct classes are 1) images of human faces and 2) everything else. This is achieved by using Haar feature-based cascade classifiers.

Original imageImage with face detection

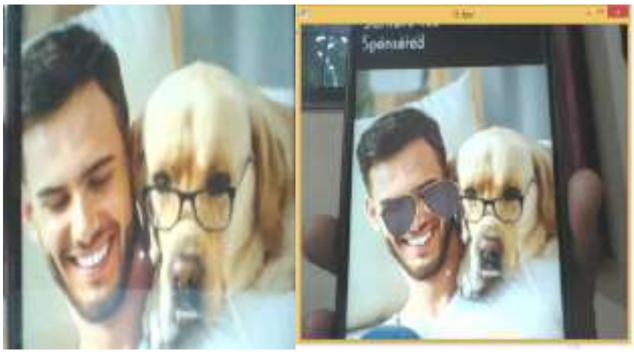

Figure 4.1Face Detection using HaarCascasde Classifier

4.1.2 Adding Eye Detection using pre-trained Haar Cascade Classifier

In this step, pre-trained Haar Cascade Classifiers are used to detect faces in an image and also to recognise the po ition of the eye in the detected face.

Figure 4.2 Eye Detection using pre-trained HaarCascade Classifier

4.2 De-noise an Image for Better Face Detection

In the context of face detection, the problem with an image like this is that - due to noise - we may miss some faces or get false detections. Using OpenCV's built in color image de-noising Volume 4 | Issue 6 | May-June 2018 | www.ijsrcseit.com 230 functionality, we de-noise the images enough so that all the faces in the image are properly detected.

4.3 Blur an Image and Perform Edge Detection

Edge detection is a dimension reduction technique - by keeping only the edges of an image we get to throw away a lot of non-discriminating information. And typically the most useful kind of edge-detection is one that preserves only the important, global structures (ignoring local structures that aren't very discriminative). So removing local structures / retaining global structures is a crucial pre-processing step to performing edge detection in an image, and blurring can do just that.

Figure 4.3 Images with Facial KeypointsDetected(using Kaggle Dataset)

V. Formulation of Modules

5.1 Modules

5.1.1 Haar Cascades:-

A face has many features that can be used to uniquely classify individuals. Using a large data store of positive and negative images, a cascade can be trained on a data store and classify Haar-like features according to the training received.

A Haar-like feature considers only adjacent spatial rectangles in an area being detected, and then giving a difference of the sum of the pixels in the white area is determined from the black area to give the relative feature.

5.1.2 Facial Landmarks:-

While Haar cascades generally have a lower computational expense in comparison to other methods, they only give approximate area of features and fail at skewed angles.

As more accurate points to accurately get the pose of the face were needed, considering facial landmarks for pin-pointing as the main method of feature detection had to be decided on.

DLIB is a modern C++ machine learning framework composing of detection algorithms and tools for solving real world problems (DLIB, n.d.). The framework recently had several new features added, including that of real-time facial landmark detection through one-millisecond facial alignment with an ensemble of regression trees.

The library quickly adds annotations of 68 facial landmarks even at non-frontal, highly skewed angles with accurate detection.

5.1.3 Anchoring:-

Anchoring landmarks to a face works by taking the mean shape of facial alignment, which allows for all images to be approximately of the same orientation.

This naturally makes sense due to the nature of the logic of more images results in better approximations as per a larger data set. Anchoring also allows for pose estimation of the head, which works to generate an approximation of the amount of available space.

5.2 Flow Diagram:-

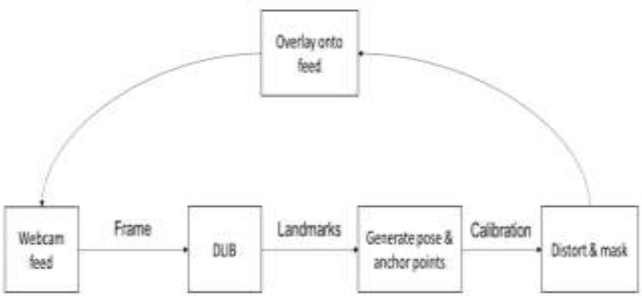

Fig. 5.1logic Flow For The Filter Overlay System

VI. Conclusion And Furtherwork

6.1 Conclusion

The OpenCV library which is used for image processing is an expanding library and hence this system is built to incorporate any changes in the library. This project would be to expand the network architecture to larger datasets, improve performance and accuracy and make the system run efficiently even on low computing devices and provide range of applications for facial biometrics.

Snapchat filters are fascinating progressions within the scope of augmented reality. The developments that have been applied to the scope of calibration and augmenting of real and fake together into a single application in the palm of one's hand has allowed for interesting deviation of reality to exist. As Snapchat grows in popularity, the quality and variations of the augmenting of what the lens captures expands constantly.

6.2 Further Work

Every application has its own merits and demerits. The project has covered almost all the requirements. Further requirements and improvements can easily be done since the coding is mainly structured or modular in nature. Changing the existing modules or adding new modules can append improvements. Further enhancements can be made to the application, so that the web site functions very attractive and useful manner than the present one.

References

[1]. "A Facial Expression Recognition System Using Convolutional Networks" by Lopes, Andre Teixeira, Edilson de Aguiar and ThiagoOliveiraSantos in 28th SIBGRAPI Conference on Graphics, Patterns and Images, IEEE(2015)

[2]. "J Learning and Transferring mid-level image representation using convolutional neural networks" by Oquab, M.Bottou, Laptev and I.Sivic in CVPR(2014)

[3]. Manske, M., 2007. Figure Visually Demonstrating the Evaluation of a Rectangle Feature with the Integral Image, s.l.: s.n.

[4]. "Facial Expression Recognition Based on TensorflowPlatform", ITM Web Conferences, 2017,

[5]. Mount, J., 2014. Approximation by orthogonal transform. [Accessed 18 April 2017].

[6]. Lienhart, R. &Maydt, J., 1998. An Extended Set of Haar-like Features for Rapid Object Detection, Santa Clara: Springer.

[7]. OpenCV, 2016. Face Detection using Haar Cascades. [Accessed 18 April 2017].