Detection of Human Motion: Adopting Machine and Deep Learning Techniques.

N.Kumaran[1]
[1]*Assistant Professor/CSE, SCSVMV University, Kanchipuram*

Abstract:-Human motion recognition has confounded the research workers on the grounds that its radical challenges. The surveillance system bounds from a regular detection of motion to understanding a complex behavior in the motion. This leads to major development in the techniques related to human motion representation and recognition. This paper discourse about the applications, general framework of human motion detection and the details of each of its components. The paper underlines on human motion representation and the recognition methods along with their merits and demerits. This study puts head together on the popular datasets and concludes with the difficulties in the domain along with a future direction. This domain has been active for more than two decades. Firstly, this presents a method for human action and spotting and classification based on multi-scale and multi-modal deep learning. Our method does not rely on labels for the real data, and no explicit transfer function is defined or learned between synthetic and real data. In our work, the data is captured by inertial sensors (such as accelerometers and gyroscopes) built in mobile devices. Having explored existing temporal models (RNN, LSTM, clockwork RNN), we show how the convolutional Clockwork RNN can be extended in a way that makes the learned features shift-invariant, and propose a more efficient training strategy for this architecture. Finally, we incorporate the learned deep features in a probabilistic biometric framework for real time user authentication.
Keywords:-Human, motion, convolutional, recognition, techniques

I. INTRODUCTION

Effective techniques for human detection are of special interest in computer vision since many applications involve people's locations and movements. Thus, significant research has been devoted to detecting, locating and tracking people in images and videos. Over the last few years the problem of detecting humans in single images has received considerable interest. Variations in illumination, shadows, and pose, as well as frequent inter- and intra-person occlusion render this a challenging task. Figure 1 shows an image of a particularly challenging scene with a large number of persons, overlaid with the results of our system. Two main approaches to human detection have been explored over the last few years. The first class of methods consists of a generative process where detected parts of the human body are combined according to a prior human model. The second class of methods considers purely statistical analysis that combine a set of low-level features within a detection window to classify the window as containing a human or not. The method presented in this paper belongs to the latter category. Recently, automated visual surveillance systems to observe certain areas are becoming more important in the research field of computer vision. Conventional surveillance systems are already installed in many areas ranging from traffic surveillance to security relevant scenarios.

However, these systems present limitations making them unsuitable in many situations. On the one hand, the systems archive huge volumes of video for eventual offline human inspection. On the other hand, security areas must be monitored by human operators, located in a control room containing a bank of screens streaming live video from each camera, for the system to be effective. CVL's contribution to visual surveillance is in the area of image sequence analysis focusing on the topics motion detection, object tracking and scene understanding:

- Motion Detection
- Object Tracking
- Scene Analysis

II. MOTION DETECTION

Motion detection algorithms are the basics for a wide range of applications in computer vision like visual surveillance, object recognition and tracking and compression of video streams. The most common approach for motion detection in surveillance systems with static cameras are the so called background subtraction algorithms. In these algorithms, a (moving) foreground object is detected by comparing the current image with the static background of the scene. The acquisition of this background image is the main challenge of background subtraction algorithms, since the background image might not be static but has to adapt to several changes as:

1. llumination changes
    - sudden changes (e.g., clouds, light-switch)

http://dx.doi.org/10.21172/ijiet.131.07

- gradual changes (e.g., position of the sun changing during the day)

2. Background motion
    - e.g., waving trees, waves
3. Changes in the background geometry
    - e.g., parking cars, moved items

Fig.1: Background subtraction method

III. OBJECT TRACKING

Object tracking can be described as a correspondence problem and involves finding which object in a video frame relates to which object in the next frame. Tracking methods can be classified into four major categories:

- Model based tracking
- Active contour-based tracking
- Feature based tracking
- Region based tracking.

Fig.2: Active contour-based tracking

IV. SCENE ANALYSIS

The aim of this type of algorithm is to recognize activities in scene. Our recognition algorithms are mainly based on statistical analysis of the scene. Rule based approaches are applied to identify e.g. abnormal behavior. The system indicates the behavior of the person.

Fig.3: Scene Analysis

V. RELATED WORK

Human detection is closely related to general object recognition techniques. It involves two steps - feature extraction and training a classifier as shown in Figure below.

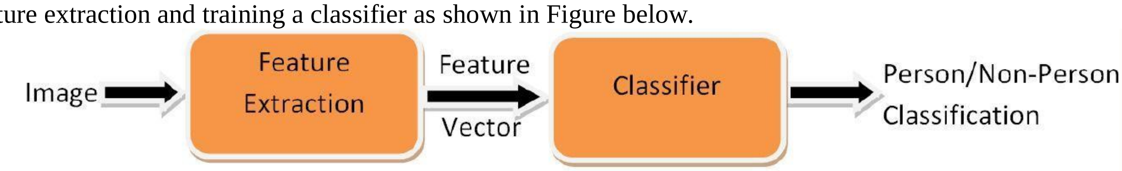

Fig.4: Components of Human Detection System.

The image feature set that needs to be extracted should be the most relevant ones for object detection or classification, while providing invariance to changes in illumination, changes in viewpoint and shifts in object contours. Such features can be based on points [1] and [2], blobs (Laplacian of Gaussian [3] or Difference of Gaussian [4]), intensities [5], gradients [6] and [7], color, texture, or combinations of several or all of these [8]. The final descriptors need to characterize the image sufficiently well for the detection and classification task at hand. We will divide the various approaches to descriptor selection into two broad categories. Sparse representations are based on local descriptors of relevant local image regions. The regions can be selected using either key point detectors, image fragments or parts detectors. On the other hand, dense representations are based on image intensities, gradients or higher order differential operators. Image features are often extracted densely (often pixel-wise) over an entire image or detection window and collected into a high-dimensional descriptor vector that can be used for discriminative image classification or labeling the window as object or non-object.

VI. EDGE DETECTION TECHNIQUES

6.1 Sobel Operator

The operator consists of a pair of 3×3 convolution kernels as shown in Table 1. One kernel is simply the other rotated by 90°

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

Table 1: Masks used by Sobel Operator

These kernels are designed to respond maximally to edges running vertically and horizontally relative to the pixel grid, one kernel for each of the two perpendicular orientations. The kernels can be applied separately to the input image, to produce separate measurements of the gradient component in each orientation (call these Gx and Gy). These can then be combined together to find the absolute magnitude of the gradient at each point and the orientation of that gradient. The gradient magnitude is given by

Typically, an approximate magnitude is computed using eq. 1 and 2.

$$|G| = \sqrt{(Gx_{2} + Gy2)} \qquad (1)$$

$$|G| = |Gx| + |Gy| \qquad (2)$$

which is much faster to compute. The angle of orientation of the edge (relative to the pixel grid) giving rise to the spatial gradient is given by eq. 3.

$$\Theta = \arctan(Gy/Gx) \qquad (3)$$

6.2 Robert's cross operator:

The Roberts Cross operator performs a simple, quick to compute, 2-D spatial gradient measurement on an image. Pixel values at point in the output represent the estimated absolute magnitude of the spatial gradient of the input image at that point. The operator consists of a pair of 2×2 convolution kernels as shown below. One kernel is simply the other rotated by 90°. This is very similar to the Sobel operator

+1	0
0	-1

Gx

0	+1
-1	0

Gy

Table 2:Masks used for Robert operator

These kernels are designed to respond maximally to edges running at 45° to the pixel grid, one kernel for each of the two perpendicular orientations. The kernels can be applied separately to the input image, to produce separate measurements of the gradient component in each orientation (call these Gx and Gy). These can then be combined together to find the absolute magnitude of the gradient at each point and the orientation of that gradient. The gradient magnitude is given by eq.4.

$$|G| = \sqrt{(Gx_{2} + Gy2)} \qquad (4)$$

although typically, an approximate magnitude is computed using eq. 5.

$$|G| = |Gx| + |Gy| \qquad (5)$$

which is much faster to compute.

The angle of orientation of the edge giving rise to the spatial gradient is given by eq. 6.

$$\Theta = \arctan(Gy/Gx) - 3\pi/4 \qquad (6)$$

Prewitt's operator:

Prewitt operator is similar to the Sobel operator and is used for detecting vertical and horizontal edges in images.

-1	0	+1
-1	0	+1
-1	0	+1

Gx

+1	+1	+1
0	0	0
-1	-1	-1

Gy

Fig: Masks for the Prewitt gradient edge detector

6.3 Laplacian of Gaussian:

The Laplacian is a 2-D isotropic measure of the 2nd spatial derivative of an image. The Laplacian of an image high lights regions of rapid intensity change and is therefore often used for edge detection. The Laplacian is often applied to an image that has first been smoothed with something approximating a Gaussian Smoothing filter in order to reduce its sensitivity to noise. The operator normally takes a single gray level image as input and produces another gray level image as output.

The Laplacian L(x, y) of an image with pixel intensity values I(x, y) is given by eq. 7.

$$L(x,y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} \qquad (7)$$

Since the input image is represented as a set of discrete pixels, we have to find a discrete convolution kernel that can approximate the second derivatives in the definition of the Laplacian. Three commonly used small kernels are shown below:

1	1	1
1	-8	1
1	1	1

-1	2	-1
2	-4	2
-1	2	-1

Table 3: Masks used for Laplacian of Gaussian

VII. EXISTING SYSTEM

Some facial recognition algorithms identify faces by extracting landmarks or features from an image. For example, an algorithm may analyze the relative position, size, and/or shape of the eyes, nose, cheekbones and jaw. These features are then used to search for other images with matching features. Other algorithms normalize a gallery of face images and then compress the face data, only saving the data in the image that is useful for face detection. A probe image is then compared with the face data. One of the earliest successful systems is based on template matching techniques applied to a set of salient facial features, providing a sort of compressed face representation. Recognition algorithms can be divided into two main approaches:

i) geometric which looks at distinguishing features (feature based) and

ii) photometric which is a statistical approach that distill an image into values and compares the values with templates to eliminate variances (view based).

VIII. PROPOSED METHOD

Previous studies have shown that significant improvement in human detection can be achieved using different types (or combinations) of low-level features. A strong set of features provides high discriminatory power, reducing the need for complex classification methods. Humans in standing positions have distinguishing characteristics. First, strong vertical edges are present along the boundaries of the body. Second, clothing is generally uniform. Clothing textures are different from natural textures observed outside of the body due to constraints on the manufacturing of printed cloth. Third, the ground is composed mostly of uniform textures. Finally, discriminatory color information is found in the face/head regions. Thus, edges, colors and textures capture important cues for discriminating humans from the background. To capture these cues, the low-level features we employ are the original HOG descriptors with additional color information, called color frequency, and texture features computed from co-occurrence matrices. To handle the high dimensionality resulting from the combination of features, PLS is employed as a dimensionality reduction technique. PLS is a powerful technique that provides dimensionality reduction for even hundreds of thousands of variables, accounting for class labels in the process. The latter point is in contrast to traditional dimensionality reduction techniques such as Principal Component Analysis (PCA). The steps performed in our detection method are the following. For each detection window in the image, features extracted using original HOG, color frequency, and co-occurrence matrices are concatenated and analyzed by the PLS model to reduce dimensionality, resulting in a low dimensional vector. Then, a simple and efficient classifier is used to classify this vector as either a human or non-human. These steps are explained in the following subsections. Flow diagram below shows a basic architecture of proposed human detection. In this propose system, images are captured using a digital camera. These images are passed through the human detection module. In the human detection module, input RGB images to convert into Gray-scale images; then normalized boundary is compared with predefined templates and if enough match is found then human is bounded by a rectangular box. After detecting human from the real-time image, the system can take several actions. Such as it can aware about the presence of the human by making alarm or displaying some light signal instructions. The global and sensational topic of the year is human detection using the closest and shortest path algorithm by binding two or more plots together.

8.1 Architecture

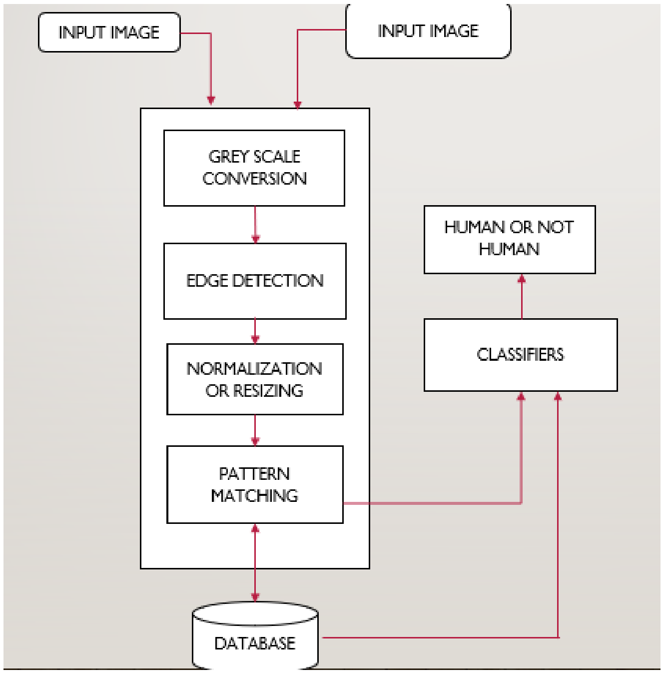

Fig 5: Architecture of the proposed system

Algorithm

1. Import necessary libraries (cv2 & imutils)
2. Save HAAR cascade files into variables by using cv2.CascadeClassifier()
3. Start capturing video by using cv2.VideoCapture().
4. Create a infinite loop
5. Read the captured video and break it into frames.
6. Resize the frame using imutil.resize().
7. Change the frame using cv2.Color() and store it into a variable gray.
8. Display the gray frame using cv2.imshow().
9. Now, by using detectMultiScale() detect the object and store the result into a variable.
10. For labeling the area use a for loop and draw rectangle and write label using cv2.rectangle() & cv2.putText() respectively.
11. Now display the final colored frame using cv2.imshow().
12. Create a break condition.
13. Release the camera and destroy the window.

IX. EXPERIMENT RESULT

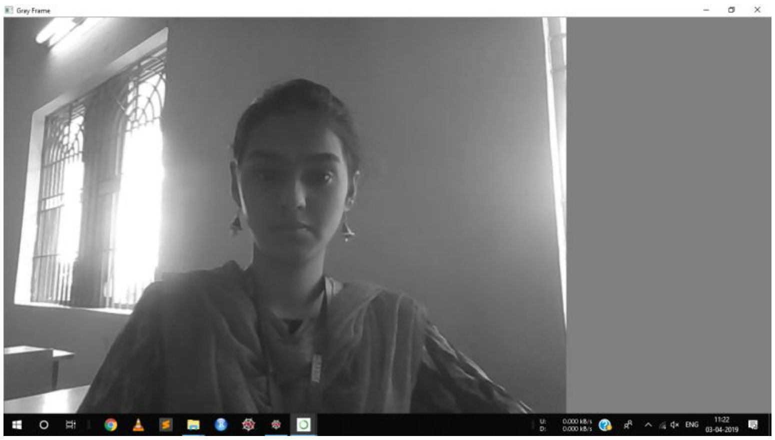

Fig 6: Gray Frame Image

http://dx.doi.org/10.21172/ijiet.131.07

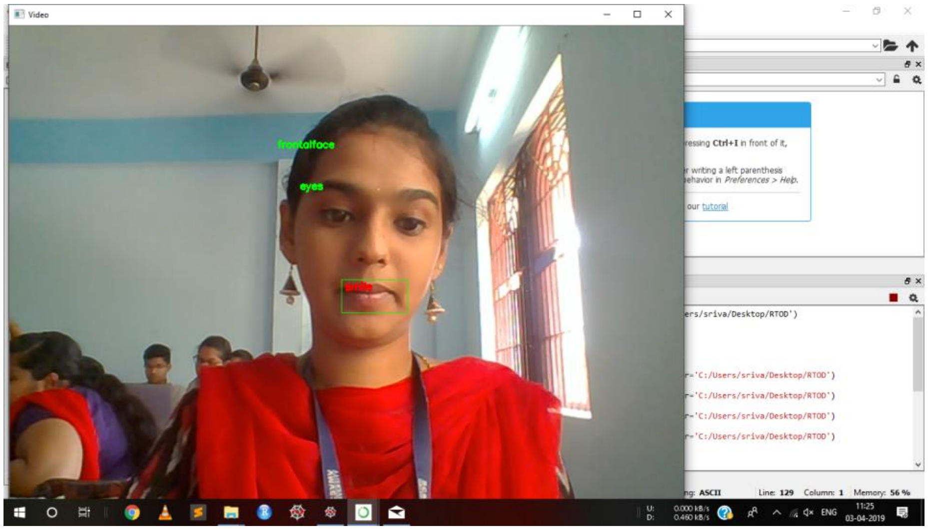

Fig 7: Human parts detection

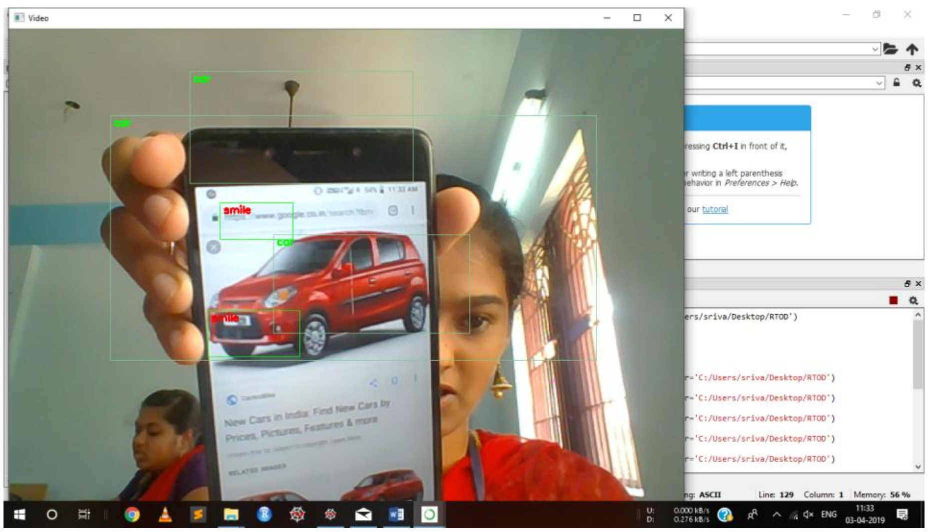

Fig 8: Car detection

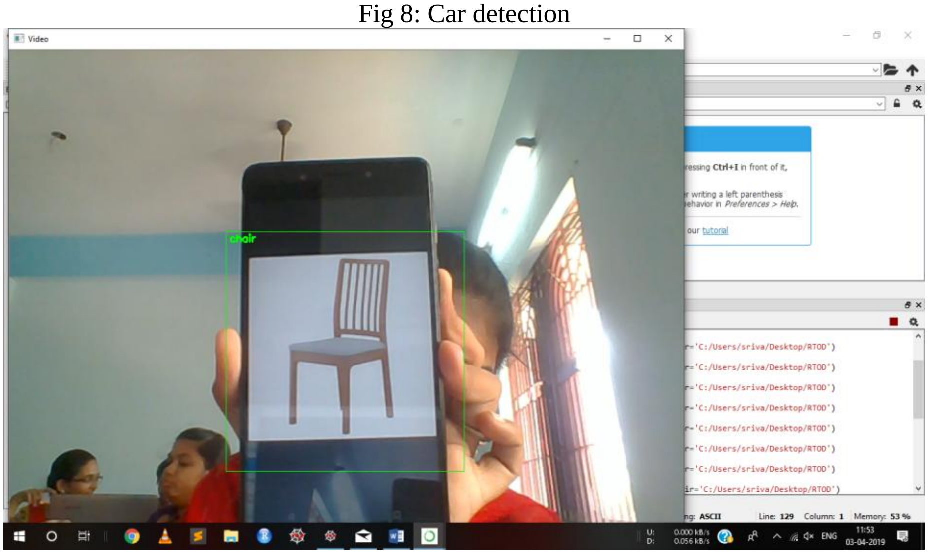

Fig 9: Detection of Object(Chair)

http://dx.doi.org/10.21172/ijiet.131.07

X. CONCLUSION AND FUTURE WORKS

This project work describes the human detection from still images using the contour/boundary-based matching. The system is tested in different position for detecting human. The proposed system runs with satisfactory success rates. The contribution of the work can be summarized as a novel method for detecting human from still images. It is a novel method for shape/contour detection from still images. It has a satisfactory performance in detecting human using contour/shape matching. Average accuracy and precision of human detection method in the system is 93.05% .

X1. REFERENCES

[1] N.Kumaran et. al., "Recognition of human actions using CNN-GWO: a novel modeling of CNN for enhancement of classification performance." Multimedia Tools and Applications(Springer) January 2018.
[2] Li Wang, Deep Learning Algorithms with Applications to Video Analytics for A Smart City: A Survey. In: IEEE 2015.
[3] Geetanjali Vinayak Kale, A Study of Vision based Human Motion Recognition and Analysis .In: International Journal of Ambient Computing and Intelligence 2016.
[4] Natalia Neverova, Deep learning for human motion analysis. Artificial Intelligence [cs.AI].In: Université de Lyon, 2016.
[5] Harris, C., Stephens., M.: A combined corner and edge detector. Alvey Vision Conference (1998).