

# HUMAN ACTIVITY FROM SURVEILLANCE CAMERA USING DEEP LEARNING

Mrs.I.Sandhya <sup>1</sup>, Nagarapu Bhavani <sup>2</sup>, V.Sravika Lakshmi <sup>3</sup>, R.Sai Chithra

<sup>1</sup> Assistant Professor, Department of Computer Science and Engineering, Sridevi Women's Engineering College, Hyderabad, India

Email: [sandhya.injeti@gmail.com](mailto:sandhya.injeti@gmail.com)

<sup>2,3,4</sup> B.Tech Student, Department of Computer Science and Engineering, Sridevi Women's Engineering College, Hyderabad, India

## ABSTRACT:

Video Surveillance plays a pivotal role in today's world. The technologies have been advanced too much when artificial intelligence, machine learning and deep learning pitched into the system. Using above combinations, different systems are in place which helps to differentiate various suspicious behaviors from the live tracking of footages. The most unpredictable one is human behaviour and it is very difficult to find whether it is suspicious or normal. Deep learning approach is used to detect suspicious or normal activity in an academic environment, and which sends an alert message to the corresponding authority, in case of predicting a suspicious activity. Monitoring is often performed through consecutive frames which are extracted from the video. The entire framework is divided into two parts. In the first part, the features are computed from video frames and in second part, based on the obtained features classifier predict the class as suspicious or normal.

## INTRODUCTION:

With increasing crime rates it becomes a problem if they are not identified in time and necessary precautionary actions taken. Most urban and metropolitan areas have surveillance systems installed which constantly accumulates data. With the vast accumulation of surveillance data there are higher chances of suspicious

activities to occur. But these tasks require human supervision to detect such activities as they are too complicated for artificial intelligence to handle and require high resources. Breaking down complicated tasks and detecting sub tasks which lead to potential crimes are one way to simplify an activity to be automated. We focus on two main potential leads

to crimes which we attempt to detect through our models.

#### **LITERATURE SURVEY:**

**Title: "A Survey of Human Activity Recognition Methods"**

**Author: Juan Ye, Xi Li, and Jiawei Wang**

Result: This survey provides an overview of various approaches to human activity recognition, including sensor-based methods, vision-based methods, and fusion-based methods. It discusses the challenges, applications, and future directions in the field.

**Title: "Recent Advances in Human Activity Recognition: A Literature Survey"**

**Author: Fatima Boujarwah and Hichem Snoussi**

Result: This survey paper summarizes recent advancements in human activity recognition techniques, including deep learning-based approaches, ensemble methods, and multimodal sensor fusion. It highlights the strengths and limitations of each method and discusses potential research directions.

**Title: "A Comprehensive Survey on Human Activity Recognition Techniques"**

**Author: Harshali Patil and Ashish Khandare**

Result: This survey provides an in-depth analysis of various human activity recognition techniques, such as machine learning algorithms, wearable sensors, and smartphone-based approaches. It discusses their performance, usability, and applications in different domains.

**Title: "Human Activity Recognition Using Wearable Sensors: A Review of Recent Trends and Challenges"**

**Author: Amir H. Behzadan and Sarah M. El-Dawy**

Result: This review paper focuses on human activity recognition using wearable sensors, discussing recent trends, challenges, and emerging technologies in the field. It also addresses issues related to data collection, preprocessing, and classification algorithms.

## **Title: "Survey on Deep Learning Techniques for Human Activity Recognition"**

**Author: Prerna Singh and Aruna Tiwari**

Result: This survey paper explores the application of deep learning techniques in human activity recognition, including convolutional neural networks (CNNs), recurrent neural networks (RNNs), and hybrid models. It discusses the advantages and limitations of deep learning approaches and identifies future research directions

### **SYSTEM ANALYSIS**

#### **EXISTING SYSTEM:**

- Artificial neural network
- Support vector machine
- Conversion
- Edge detection

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning. The goal of the

SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane. SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine. Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane

#### **DRAWBACKS :**

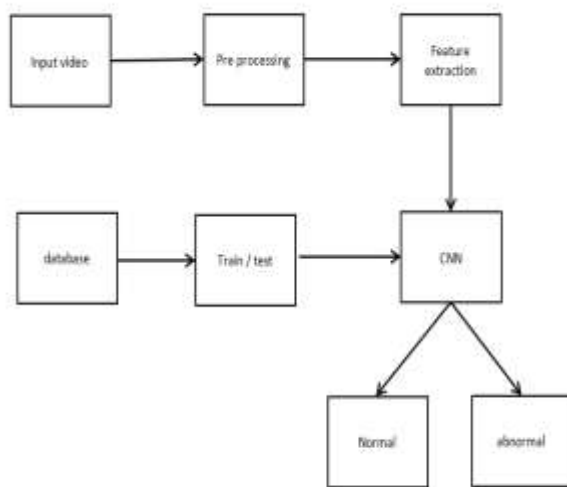
- Less accuracy
- Training part is less .

#### **PROPOSED SYSTEM:**

- Input video
- Pre-processing
- Feature extraction
- CNN (convolutional neural network)

**ADVANTAGE:**

- High accuracy
- It will identify the suspect with videos

**SYSTEM DESIGN****SYSTEM ARCHITECTURE:****IMPLEMENTATION MODULES:**

- Video streaming: Video streaming technology is one way to deliver video over the Internet. Using streaming

technologies, the delivery of audio and video over the Internet can reach many millions of customer using their personal computers, PDAs, mobile smartphones or other streaming devices.

- Pre-processing :in this step we have to reduce the complexity of the picture of license plate using resize and the conversion.using these pre-process we can change the size of the license plate using resize.
- Blob detection:A Blob is a group of connected pixels in an image that share some common property ( E.g grayscale value ). In the image above, the dark connected regions are blobs, and the goal of blob detection is to identify and mark these regions.
- Deep learning:Within Deep Learning, a Convolutional Neural Network or CNN is a

type of artificial neural network, which is widely used for image/object recognition and classification. Deep Learning thus recognizes objects in an image by using a CNN.

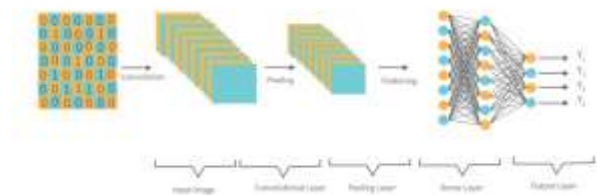
## ALGORITHMS:

## CONVOLUTION NEURAL NETWORK

### INTRODUCTION

Convolutional neural networks. Sounds like a weird combination of biology and math with a little CS sprinkled in, but these networks have been some of the most influential innovations in the field of computer vision. 2012 was the first year that neural nets grew to prominence as Alex Krizhevsky used them to win that year's ImageNet competition (basically, the annual Olympics of computer vision), dropping the classification error record from 26% to 15%, an astounding

improvement at the time. Ever since then, a host of companies have been using deep learning at the core of their services. Facebook uses neural nets for their automatic tagging algorithms, Google for their photo search, Amazon for their product recommendations, Pinterest for their home feed personalization, and Instagram for their search infrastructure.



### The Problem Space

Image classification is the task of taking an input image and outputting a class (a cat, dog, etc) or a probability of classes that best describes the image. For humans, this task of recognition is one of the first skills we learn from the moment we are born and is one that comes naturally and effortlessly as adults. Without even

thinking twice, we're able to quickly and seamlessly identify the environment we are in as well as the objects that surround us. When we see an image or just when we look at the world around us, most of the time we are able to immediately characterize the scene and give each object a label, all without even consciously noticing. These skills of being able to quickly recognize patterns, generalize from prior knowledge, and adapt to different image environments are ones that we do not share with our fellow machines.

### Inputs and Outputs

When a computer sees an image (takes an image as input), it will see an array of pixel values. Depending on the resolution and size of the image, it will see a  $32 \times 32 \times 3$  array of numbers (The 3 refers to RGB values). Just to drive home the point, let's say we have a color image in JPG form and its size is  $480 \times 480$ . The representative array will be  $480 \times 480 \times 3$ . Each of

these numbers is given a value from 0 to 255 which describes the pixel intensity at that point. These numbers, while meaningless to us when we perform image classification, are the only inputs available to the computer. The idea is that you give the computer this array of numbers and it will output numbers that describe the probability of the image being a certain class (.80 for cat, .15 for dog, .05 for bird, etc).

### What We Want the Computer to Do

Now that we know the problem as well as the inputs and outputs, let's think about how to approach this. What we want the computer to do is to be able to differentiate between all the images it's given and figure out the unique features that make a dog a dog or that make a cat a cat. This is the process that goes on in our minds subconsciously as well. When we look at a picture of a dog, we can classify it as such if the picture has identifiable features such as

paws or 4 legs. In a similar way, the computer is able to perform image classification by looking for low level features such as edges and curves, and then building up to more abstract concepts through a series of convolutional layers. This is a general overview of what a CNN does. Let's get into the specifics.

### Biological Connection

But first, a little background. When you first heard of the term convolutional neural networks, you may have thought of something related to neuroscience or biology, and you would be right. Sort of. CNNs do take a biological inspiration from the visual cortex. The visual cortex has small regions of cells that are sensitive to specific regions of the visual field. This idea was expanded upon by a fascinating experiment by Hubel and Wiesel in 1962 (Video) where they showed that some individual neuronal cells in the visual cortex responded (or fired) only in the presence of edges of a certain orientation. For example, some neurons fired when exposed to vertical edges and some when shown horizontal or diagonal edges. Hubel and Wiesel found out that all of these neurons were organized in a columnar architecture and that together, they were able to produce visual perception. This idea of specialized components inside of a system having specific tasks (the neuronal cells in the visual cortex looking for specific characteristics) is one that machines use as well, and is the basis behind CNNs.

Structure

### Structure

Back to the specifics. A more detailed overview of what CNNs do would be that you take the image, pass it through a series of convolutional, nonlinear, pooling (downsampling), and fully connected layers, and get an output. As we said earlier, the output can be a single class or a probability of

classes that best describes the image. Now, the hard part is understanding what each of these layers do. So let's get into the most important one.

### First Layer – Math Part

The first layer in a CNN is always a Convolutional Layer. First thing to make sure you remember is what the input to this conv (I'll be using that abbreviation a lot) layer is. Like we mentioned before, the input is a  $32 \times 32 \times 3$  array of pixel values. Now, the best way to explain a conv layer is to imagine a flashlight that is shining over the top left of the image. Let's say that the light this flashlight shines covers a  $5 \times 5$  area. And now, let's imagine this flashlight sliding across all the areas of the input image. In machine learning terms, this flashlight is called a filter (or sometimes referred to as a neuron or a kernel) and the region that it is shining over is called the receptive field. Now this filter is also an array of numbers (the

numbers are called weights or parameters). A very important note is that the depth of this filter has to be the same as the depth of the input (this makes sure that the math works out), so the dimensions of this filter is  $5 \times 5 \times 3$ . Now, let's take the first position the filter is in for example. It would be the top left corner. As the filter is sliding, or convolving, around the input image, it is multiplying the values in the filter with the original pixel values of the image (aka computing element wise multiplications). These multiplications are all summed up (mathematically speaking, this would be 75 multiplications in total). So now you have a single number. Remember, this number is just representative of when the filter is at the top left of the image. Now, we repeat this process for every location on the input volume. (Next step would be moving the filter to the right by 1 unit, then right again by 1, and so on). Every unique location on the input volume produces a number.

After sliding the filter over all the locations, you will find out that what you're left with is a  $28 \times 28 \times 1$  array of numbers, which we call an activation map or feature map. The reason you get a  $28 \times 28$  array is that there are 784 different locations that a  $5 \times 5$  filter can fit on a  $32 \times 32$  input image. These 784 numbers are mapped to a  $28 \times 28$  array.

#### First Layer – High Level Perspective

However, let's talk about what this convolution is actually doing from a high level. Each of these filters can be thought of as feature identifiers. When I say features, I'm talking about things like straight edges, simple colors, and curves. Think about the simplest characteristics that all images have in common with each other. Let's say our first filter is  $7 \times 7 \times 3$  and is going to be a curve detector. (In this section, let's ignore the fact that the filter is 3 units deep and only consider the top depth slice of the filter and the

image, for simplicity.) As a curve detector, the filter will have a pixel structure in which there will be higher numerical values along the area that is a shape of a curve (Remember, these filters that we're talking about as just numbers!).

#### **SOFTWARE ENVIRONMENT:**

##### PYTHON LANGUAGE:

- Python idle
- Anaconda navigator
- opencv

#### **HARDWARE REQUIREMENTS:**

- 1) Operating System : Windows Only
- 2) Processor : i5 and above
- 3) Ram : 4gb and above
- 4) Hard Disk : 50 GB

#### **TEST CASES:**

Test case1:(packages testing)

Input: downloading packages in interactive mode

Output: importing packages in script mode

Output: do the training part with algorithm and check live video detecting normal / abnormal

Test case2: (IDLE testing)

Input : user execution in IDLE

Output: Ip camera in command prompt

Test case3:(data process)

Input : load data

Output: load data and display data in output code

Test case 4:(pre-process)

Input: do pre-process

Output: did pre-process using resize and conversion

Test case 6:(output)

Input : find output



### **FUTURE SCOPE:**

The future of detecting suspicious human activity from surveillance cameras using deep learning holds significant promise, with opportunities for continued advancements in accuracy, real-time detection, privacy preservation, and robustness against adversarial attacks.

### **CONCLUSION**

We created an CNN model for detecting activity like fighting, walking ,jumping & running from CCTV footage, the model was trained

on videos and achieved an good accuracy .

## REFERENCE:

[1] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards RealTime Object Detection with Region Proposal Networks," in NIPS 2015 - Advances in Neural Information Processing Systems 28. Neural Information Processing Systems Foundation, 2015

[2] Pierre Sermanet David Eigen Xiang Zhang Michael Mathieu Rob Fergus Yann LeCun "OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks" Courant Institute of Mathematical Sciences, New York University. Available: <https://arxiv.org/pdf/1312.6229.pdf>.

[3] The IMFDB Internet Movie Firearms Database, [Online] Available: [http://www.imfdb.org/wiki/Main\\_Page](http://www.imfdb.org/wiki/Main_Page) [Accessed Mar 20, 2019]

[4] Soft Computing and Intelligence Information Systems, [Online] Available: <https://sci2s.ugr.es/weapons-detection> [Accessed Mar 27, 2019]

[5] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna,

"Rethinking the Inception Architecture for Computer Vision," in arXiv, vol. abs/1512.00567, 2015

[6] Rohit Kumar Tiwari and Gyanendra K. Verma, "A Computer Vision based Framework for Visual Gun Detection using Harris Interest Point Detector", Procedia Computer Science, vol 54, p. 703 - 712, 2015.

[7] Samir K. Bandyopadhyay, Biswajita Datta, and Sudipta Roy Identifications of concealed weapon in a Human Body Department of Computer Science and Engineer, University of Calcutta, 2012

[8] Claire-Hlne Demarty, et. al The MediaEval 2012 affect task: violent scenes detection Working Notes Proceedings of the MediaEval 2012 Workshop. 2012.

[9] Roberto Olmos, Siham Tabik, and Francisco Herrera Automatic Handgun Detection Alarm in Videos Using Deep Learning Soft Computing and Intelligent Information Systems research group, Department of Computer Science and Artificial Intelligence, University of Granada, 2017

[10] Lisa Torrey and Jude Shavlik, "Transfer Learning", University of

Wisconsin, Madison WI, USA,  
Available:  
<http://ftp.cs.wisc.edu/machine->

[learning/shavlikgroup/torrey.handbook09.pdf](#) [Accessed Mar 15, 2019]