Surveillance

Aishwarya Phirke (C-44), Amartya Mathew (C-50), Neha Khare (C-68)
Department of Computer Engineering
Vishwakarma Institute of Technology, Pune.

Nikhil Kulkarni (H-24), Raghav Sadany (H-09)
Department of Information Technology
Vishwakarma Institute of Technology, Pune.

Sumit Tanpure (V-46)
Department of Mechanical Engineering
Vishwakarma Institute of Technology, Pune.

Abstract –A prototype of a novel Surveillance system whichinvolves mounting cameras on an unmanned terrain vehicle. The cameras on board will detect motion and try to classify what has caused the motion using image processing. This will be a good tool for surveillance in areas where having human personal is risky or not possible.

Index Terms *- Motion Detection, Object Recognition, Communication, Android Interface, Convolutional Neural Network, Optical Flow.*

I. Introduction

Country borders and war areas yield inevitable injuries and fatalities with most recipients recognized for their outstanding devotion and commitment to country. Scientists have studied many animal characteristics and have trained some of them for certain tasks that range widely from force protection to military applications [1][2][3]. We propose a prototype of anovel Surveillance system which involves mounting cameras on an unmanned terrain vehicle that takes video inputs andsends them to the system. The system detects motion from a streamed video file and analyses the frames in real-time, required that various aspects to be taken into consideration. Some of these aspects are in contrast created by weather outside, flying machines that may fly over a restricted zone and strong winds that can affect stationary objects to activate the system.

So instead of a person performing surveillance we can mount a camera on any unmounted vehicle and by image processing data required can be obtained. The System functioning has four major steps:

1 Optical Flow(Motion Detection)
2 Convolutional Neural Network(Image Classification)
3 Firebase(Storage/Accessing Data for Remaining Operations)
4 Android Application(For the User to have access to the data with a better interface)

II. THEORY

A. Optical Flow

Optical flow is the pattern of apparent motion of image objects between two consecutive frames caused by the movement of object or camera. It is 2D vector field where each vector is a displacement vector showing the movement of points from first frame to second as seen in fig.1.

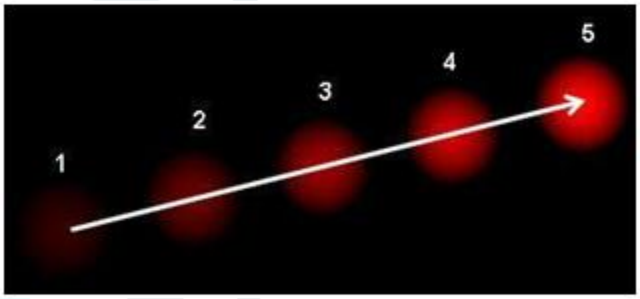

Fig.1 Optical flow frames. Source: image1

Fig.1 shows a ball moving in 5 consecutive frames. The arrow shows its displacement vector. Optical flow has many applications in areas like:

1 Structure from Motion
2 Video Compression
3 Video Stabilization

Optical flow works on several assumptions:

1. The pixel intensities of an object do not change between consecutive frames.
2. Neighbouring pixels have similar motion.

B. Convolutional Neural Network(CNN)

Convolutional Neural Networks (ConvNets or CNNs) are a category of Neural Networks that have proven very effective in areas such as image recognition and classification. ConvNets have been successful in identifying faces, objects and traffic signs apart from powering vision in robots and self driving cars[4].
ConvNet is able to recognize scenes and the system is able to suggest relevant captions (“a soccer player is kicking a soccer ball”) while Figure below shows an example of ConvNets being used for recognizing everyday objects, humans and animals. Lately, ConvNets have been effective in several

Natural Language Processing tasks (such as sentence classification) as well.

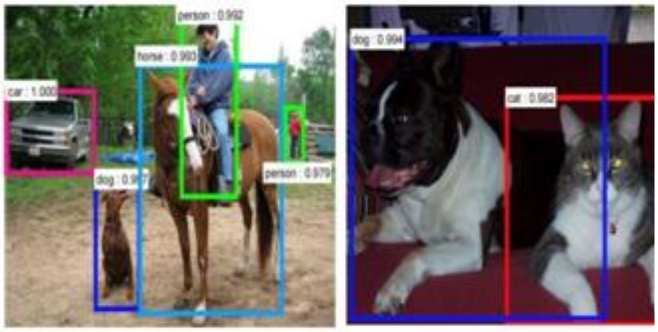

Fig.2 ConvNet recognized images. Source : image2

The four main operations in the ConvNet are:
1)Convolution
2) Non Linearity (ReLU)
3) Pooling or Sub Sampling
4) Classification (Fully Connected Layer)

C. Firebase

Firebase Storage provides secure file uploads and downloads for Firebase apps, regardless of network quality. The developer can use it to store images, audio, video, or other user-generated content. Firebase Storage is backed by Google

Cloud Storage. The Data obtained from Optical flow is the processing Data for CNN hence to get access we have used Firebase as storage where the further data will be stored as well. Similarly like CNN, the Android Application would get access to Firebase and would display them in a friendly user interface.

D. Android

Android Application is the last stage of the Task where the user has access to the required data only. From the motion being detected to the classification of the Motion and then finally to the Android Application presented in a better and less complicated interface.

III. Algorithm

1. 221 equidistant pixels (17*13) in the video frame are set to be considered for optical flow each being 20 pixels apart.
2. Every successive frame in the video gives the displacement of those 221 points in the new image using optical flow.
3. We add the values of the average of all the displacement into a queue of size 10.
4. Using this the threshold after every 10 frames is automatically refreshed.

   final_threshold = final_thresh/10
5. Now, a different queue is created to add all the displacements of 221 points of every frame to check their displacement with respect to the automated threshold value received before

   threshP = LP.get()
6. We set the minimum and maximum number of points required for detection.
7. We do this to get accurate results and reduce the unwanted noise due to various factors.
8. The detected images are now sent to the firebase.

IV Workflow

The Threshold process is done automatically as it takes an **average of the first ten frames** and applied the value to thenext ten frames and the process goes on.

Cause of automating the threshold it becomes easier to upload the code on any device without adjusting the speed of the camera and accuracy also increases.

The calculation for finding out the difference in the Frames is as follows:

Consider a pixel $I(x,y,t)$ in first frame.A new dimension, time, is added here. Extra time is not needed as earlier only images were considered It moves by distance (dx,dy) in next frame taken after dt time. So since those pixels are the same and intensity does not change, we can say,

$$I(x,y,t)=I(x+dx,y+dy,t+dt)$$

Taking taylor series approximation of right-hand side, remove common terms and divide by *dt* to get the following equation:

$$fxu+fyv+ft=0$$

where,

$$\boldsymbol{fx=\partial f\partial x;fy=\partial f\partial y}$$

$$u=dxdt;v=dydt \qquad \textbf{……(1)}$$

Above equation (1) is called Optical Flow equation. We can find *fx* and *fy*, they are image gradients.

Similarly *ft* is the gradient along time. As *(u,v)* is unknown, this one equation with two unknown variables is unsolvable.

B. Image Classification

The images uploaded on Firebase are accessed for their classification. By using CNN we first train the machine with a Dataset using Back propagation for recognizing and classifying them. This trained code is then used in classifying the images of firebase .If the images are of concern to the user then it is labelled and uploaded on Firebase according to the categories for example human, animal, machines etc.[5][6]

A. Motion Detection

The Python program for Motion Detection is responsible for the data extracted. When any object is in motion faster or slower than the set threshold of the code then that particular frame is extracted and stored in the form of an image. This image is transferred on the online storage platform used Firebase.

The work flow of the code is such that, it compares two frames and plots fixed points on each frame. The change in positioning of the points helps us in calculating in the displacement of the points.

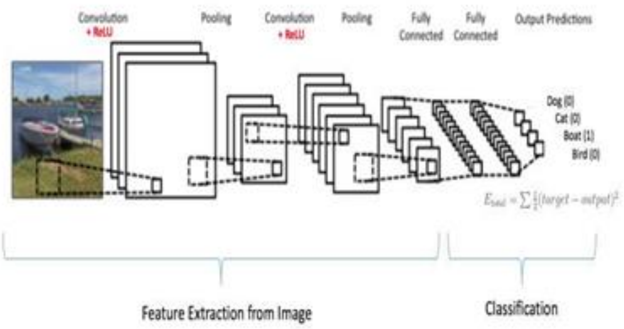

Fig.3 Object classification Flow chart Source: image3

The overall training process of the Convolution Network may be summarized as below:
Step1: We initialize all filters and parameters / weights withrandom values.

Step2: The network takes a training image as input, goesthrough the forward propagation step (convolution, ReLU and pooling operations along with forward propagation in the Fully Connected layer) and finds the output probabilities for each class.

Assuming the output probabilities for the boat image above are [0.2, 0.4, 0.1, 0.3]

Since weights are randomly assigned for the first training example, output probabilities are also random.

Step3: Calculate the total error at the output layer (summationover all 4 classes)

Total Error = ∑ ½ ***(target probability – output probability)*** ²

Step4: Use Back propagation to calculate the*gradients*of theerror with respect to all weights in the network and use *gradient descent* to update all filter values / weights and parameter values to minimize the output error.

1. The weights are adjusted in proportion to their contribution to the total error.
2. When the same image is input again, output probabilities might now be [0.1, 0.1, 0.7, 0.1], which is closer to the target vector [0, 0, 1, 0].
3. This means that the network has *learnt* to classify this particular image correctly by adjusting its weights / filters such that the output error is reduced.
4. Parameters like number of filters, filter sizes, architecture

   of the network etc. have all been fixed before Step 1 and do not change during training process – only the values of the filter matrix and connection weights get updated.

Step5: Repeat steps 2-4 with all images in the training set.The above steps *train* the ConvNet – this essentially means that all the weights and parameters of the ConvNet have now been optimized to correctly classify images from the training set.

When a new (unseen) image is input into the ConvNet, the network would go through the forward propagation step and output a probability for each class (for a new image, the output probabilities are calculated using the weights which have been optimized to correctly classify all the previous training examples). If our training set is large enough, the network will (hopefully) generalize well to new images and classify them into correct categories.

C. *Android and Firebase interface*

After the classification is done the images are again posted on Firebase. The Android Application made directly gets access to the images posted on firebase with its simple and user friendly interface. The Application is made using Java and Android Studio.

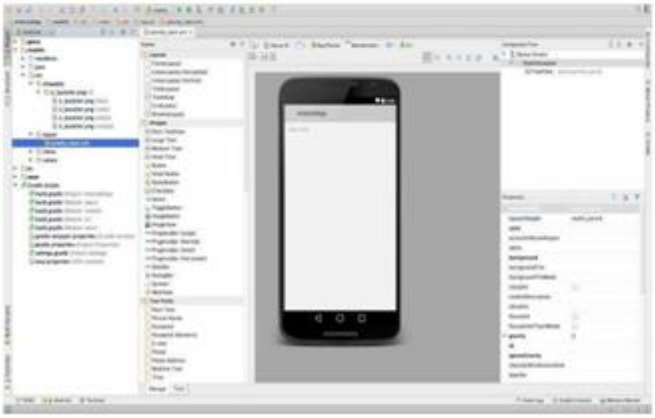

Fig.4 Android interface Source: Internet

V FUTURE SCOPE

1. The code can be run on Raspberry Pi module 3. It has an inbuilt WIFI module on it. A camera can be also mounted on it and a drone (bot/air drone) can be controlled using Rpi. The only problem is that there is a hardware restriction. The RAM of Rpi is not sufficient enough to run the Program in a sufficient way. It is very slow as compared to that on Computer.

Fig.5 Raspberry Pi module 3. Source: Internet

2. A vast area as Computer Vision can be used as data providing source in many applications. It has a huge scope in Surveillance as well cause of automating the whole monitoring system.
3. Better communication techniques can be used like GSM Module.
4. GPS can be used for the accurate coordinates for locating the threats faster and efficiently using the available resources.
5. The blackout timings can be detected so that there is no threat during that period.

VI CONCLUSION

Surveillance has been implement without the presence of any Human. This concept can be used in many fields to recognize thing where a continuous monitoring is required. Making such things automated make the chances of Errors very less. For current time we are not able to mount it on an unmanned drone due to Hardware restrictions.

VII ACKNOWLWDGEMENT

We would like to thank our project mentors Prof. S. G. Lade, Department of Computer Engineering, Vishwakarma Institute of Technology; Prof.D.R.Deshpande, Department of Information Technology, Vishwakarma Institute of Technology and Prof.A.R.Mujumdar, Department of Mechanical Engineering, Vishwakarma Institute of Technology for their invaluable guidance and kind cooperation.

REFERENCES

[1]. J. Khurshid, “Military robots – a glimpse from today and tomorrow” 2004.

[2]. A.Nayyar, “Smart Surveillance Robot for Real-Time Monitoring and Control System in Environment and Industrial Applications” 2018.

[3]. S. Joshi , “Surveillance robot for military application,” 2016.

[4]. Widodo Budiharto”Intelligent Surveillance Robot with Obstacle Avoidance Capabilities Using Neural Network,” 2015.

[5]. M.Karthikeyan, “Intelligent Exploration and Surveillance Robot In Defence Environment” 2014.

[6]. S.Nalawade,”Robots For Surveillance In Mitary Applications” 2014.