International Journal of Advanced Development in Science and Technology
Volume :3 Issue : 03 April 2021 www.ijadst.com e-ISSN: 2582-1059

SENIOR PEOPLE SAFETY MONITORING AND ALERT SYSTEM

[1] K.Aravindhan, [2] T.Balamurugan, [3] R.K. Kapilavani, [4] B. Umamaheswari
[1,2] Students, [3,4] Faculty,
[1,2 3,4] Prince Shri Venkateshwara Padmavathy Engineering College, Ponmar,
aravindhankumarappa@gmail.com.

ABSTRACT

In the present scenario one of the common diseases that is found in elderly people is dementia.Dementia is a brain disorder that most often affects the elderly.Monitoring and taking care of these people is been a major factor. This project showcases a smart human action recognition method to automatically identify the human activities from skeletal joint motions and combines the competencies. This is a low-cost method and has high accuracy.An independent mobile application is also developed to monitor the condition of the people and its surroundings when they are alone. A Notification API integration facilitates sending alert notification during abnormal condition is also implemented in the mobile application. Thus, our project provides a way to help the senior citizens and children from any kind of mishaps and health issues. Thus, our project provides a way to help the senior citizens and children from any kind of mishaps and health issues.

Keywords- mobile application, API Integration.

1. INTRODUCTION

Human activity recognition, or HAR, is a challenging time series classification task. It involves predicting the movement of a person by deep domain expertise and methods from signal processing to correctly engineer features from the raw data in order to fit a machine learning model.

Recently, deep learning methods such as convolutional neural networks and recurrent neural networks have shown capable and even achieve state-of-the-art results by automatically learning features from the data collected.

In this post, you will discover the problem of human activity recognition and the deep learning methods that are achieving state-of-the-art performance on this problem.

After reading this post, you will know:

- Activity recognition is the problem of predicting the movement of a person, often indoors, based on collected data, such as an accelerometer in a smartphone.
- Streams of the datasets are often split into subs-sequences called windows,

and each window is associated with a broader activity, called a sliding window approach.

- Convolutional neural networks and long short-term memory networks, and perhaps both together, are best suited to learning features from collected data and predicting the associated movement.

Human activity recognition, or HAR for short, is a broad field of study concerned with identifying the specific movement or action of a person based on data. Movements are often typical activities performed indoors, such as walking, sleeping, standing, and sitting. They may also be more focused activities such as those types of activities performed in a kitchen or on a factory floor. The data may be remotely recorded, such as video, or other wireless methods. Alternately, data may be recorded directly on the subject such as by carrying custom hardware or smart phones that have accelerometers and gyroscopes.

Historically, data for activity recognition was challenging and expensive to collect, requiring custom hardware. Now smart phones and other personal tracking devices used for fitness and health monitoring are cheap and ubiquitous. As such, data from these devices is cheaper to collect, more common, and therefore is a more commonly studied version of the general activity recognition problem.

The problem is to predict the activity given a snapshot of data. Generally, this problem is framed as a univariate or multivariate time series classification task. It is a challenging problem as there are no obvious or direct ways to relate the recorded data to specific human activities and each subject may perform an activity with significant variation, resulting in variations in the recorded data. The intent is to record data and corresponding activities for specific subjects, fit a model from this data, and generalize the model to classify the activity of new unseen subjects from their data.

2. RELATED WORK

According to Suraj Prakash Sahoo, Samit Ari, Saraju P. Mohanty Human action recognition (HAR) is a challenging task due to the presence of the pose and temporal variations in the action videos. To address these challenges, HAR-Depth is proposed in this paper with sequential and shape learning along with the novel concept of depth history image (DHI). A deep bidirectional long short termmemory (DBiLSTM) is constructed for sequential learning to model the temporal relationship existing between the action frames. Action information in each frame is extracted using pre-trained convolutional neural network (CNN). The depth information of each action frame is estimated and projected onto the X-Y plane to form the DHI. During shape learning, the shape information through DHI is used to train a deep pre-trained CNN network. By leveraging the trained knowledge of the pre-trained network, overfitting issue is handled. The finetuned network is used to recognize actions from query DHI images. Data augmentation is adopted to avoid overfitting of the network by virtually increasing the training set. The proposed work is evaluated on publicly available datasets like KTH, UCF sports, JHMDB, UCF101, and HMDB51 and achieves the performance accuracy of 97.67%, 95.00%, 73.13%, 92.97%, and 69.74% respectively. The results on these datasets suggest that the proposed work of this paper performs better in terms of overall accuracy, kappa parameter and precision compared to the other state-of-the-art algorithms present in the earlier reported literature.

According to Lei Wang, Xu Zhao, Yunfei Si, Liangliang Cao, Yuncai Liu Human activity recognition is a challenging high level vision task, for which multiple factors, as subject, object and their diverse interactions, have to be considered and modelled. Current learning based methods are limited in the capability to integrate human level concepts into an easily extensible computational framework.

Inspired by the existing human memory model, we present a context-associative approach to recognize activity with human-object interaction. The proposed system can recognize incoming visual content based on the previous experienced activities. The high-level activity is parsed into consecutive sub-activities, and we build a context cluster to model the temporal relations.

The semantic attributes of the sub-activity are organized by a concept hierarchy. Based on the hierarchy, a series of similarity functions is defined to turn the recognition computing into retrievals over the contextual memory, similar to the auto-associative characteristics of human memory. Partially matching in retrieval and stored memory make the activity prediction possible. Brain memory's dynamical evolution is mimicked to allow decay and reinforcement of the input information, providing a natural way to maintain data and save computational time. We evaluate our approach on three data sets, CAD-120, MHOI and OPPORTUNITY. The proposed method demonstrates promising results compared with other state-of-the-art techniques.

3. PROBLEM DEFINITION

Monitoring the elderly is becoming difficult whenthey travel around or left alone in home. Situation like this put the people in problem to take care of their near and dear ones.

3.1 SYSTEM MODEL

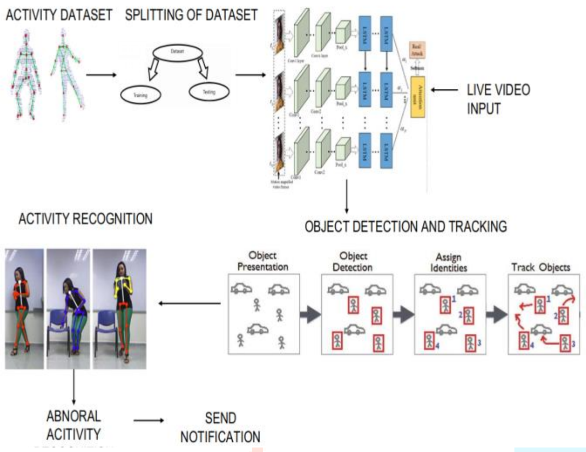


Fig1: Architecture Diagram

3.1.1 DATASET COLLECTION

A data set is a collection of data. Deep Learning has become the go-to method for solving many challenging real-world problems. It's definitely by far the best performing method for computer vision tasks. The image above showcases the power of deep learning for computer vision. With enough training, a deep network can segment and identify the "key points" of every person in the image. These deep learning machines that have been working so well need fuel lots of fuel; that fuel is data. The more labelled data available, the better our model performs. The idea of more data leading to better performance has even been explored at a large-scale by Google with a dataset of 300 Million images! When deploying a Deep Learning model in a real-world application, data must be constantly fed to continue improving its performance. And, in the deep learning era, data is very well arguably the most valuable resource. There are three steps of collecting data

3.1.2 SCRAPING FROM THE WEB

Manually finding and downloading images takes a long time simply due to the amount of human work involved. The task probably has some kind of common objects are to be detected. And so that becomes the keyword for web-scraping. It also becomes the class name for that object. From the sounds of it this is of course very easy for a task such as image classification where the images annotations are quite coarse. But to do something like instance segmentation? Every single pixel in the image is required. To get those, it's best to use some really great image annotation tools that are already out there. The paper shows how to create a model that, given a rough set of polygon points around an object, can generate the pixel labels for segmentation. Deep extreme cut is also quite similar except they use only the four extreme points around the object. This will then give some nice bounding box and segmentation labels. Another option is to use an existing image annotation GUIs. Label someone very popular where one can draw both bounding boxes and set polygon points for segmentation maps. Amazon Mechanical Turk is also a cheap option.

3.1.3 THIRD-PARTY

Since data has become such a valuable commodity in the deep learning era, many start-ups have started to offer their own image annotation services they'll gather and label the data. Given a description of what kind of data and annotations needed. Mighty is one that has been doing self-driving car image annotation and has become pretty big in the space were at CVPR 2018 too. Payment AI are less specialized than Mighty AI, offering image annotation for any domain. They also offer a couple more tools such as video and landmark annotations.

3.1.4 DATASET AUGMENTATION

The performance of deep learning neural networks often improves with the amount of data available. Data augmentation is a technique to artificially create new training data from existing training data. This is done by applying domain-specific techniques to examples from the training data that create new and different training examples. Image data augmentation is perhaps the most well-known type of data augmentation and involves creating transformed versions of images in the training dataset that belong to the same class as the original image. Transforms include a range of operations from the field of image manipulation, such as shifts, flips, zooms, and much more. The intent is to expand the training dataset with new, plausible examples. This means, variations of the training set images that are likely to be seen by the model. For example, a horizontal flip of a picture of a cat may make sense, because the photo could have been taken from the left or right. A vertical flip of the photo of a cat does not make sense and would probably not be appropriate given that the model is very unlikely to see a photo of an upside down cat. As such, it is clear that the choice of the specific data augmentation techniques used for a training dataset must be chosen carefully and within the context of the training dataset and knowledge of the problem domain. In addition, it can be useful to experiment with data augmentation methods in isolation and in concert to see if they result in a measurable improvement to model performance, perhaps with a small prototype dataset, model, and training run. Modern deep learning algorithms, such as the convolutional neural network, or CNN, can learn features that are invariant to their location in the image. Nevertheless, augmentation can further aid in this transform invariant approach to learning and can aid the model in learning features that are also invariant to transforms such as left-to-right to top-to-bottom ordering, light levels in photographs, and more. Image data augmentation is typically only applied to the training dataset, and not to the validation or test dataset. This is different from data preparation such as image resizing and pixel scaling; they must be performed consistently across all datasets that interact with the model.

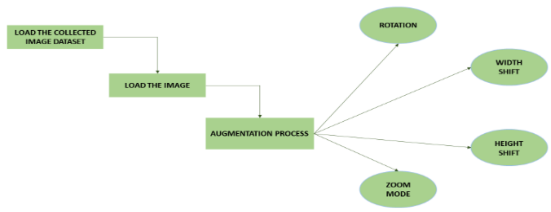


Fig2: Data Augmentation

3.1.5 DATASET PREPROCESSING MODULE

Deep learning has truly come into the mainstream in the past few years. Deep learning uses neural nets with a lot of hidden layers (dozens in today's state of the art) and requires large amounts of training data. These models have been particularly effective in gaining insight and approaching human-level accuracy in perceptual tasks like vision, speech, language processing. The theory and mathematical foundations were laid several decades ago. Primarily two phenomena have contributed to the rise of machine learning a) Availability of huge data-sets/training examples in multiple domains and b) Advances in raw compute power and the rise of efficient parallel hardware.

Building an effective neural network model requires careful consideration of the network architecture as well as the input data format. This article deals with the latter. The most common image data input parameters are the number of images, image height, image width, number of channels, and the number of levels per pixel. Typically we have 3 channels of data corresponding to the colors Red, Green, Blue (RGB) Pixel levels are usually [0,255]. For this exercise let's choose the following values

- Number of images = 100
- Image width, image height =100
- 3 channels, pixel levels in the range [0–255]

Labeled Faces in the Wild is a database of facial images, originally designed for studying the problem of face recognition. The data-set contains more than 13,000 images of faces collected from the web, and each face has been labeled with the name of the person pictured. Data-set images need to be converted into the described format. After downloading the image data, notice that the images are arranged in separate sub-folders, by name of the person. We'll need to get all the photos into a common directory for this exercise. Let's take the first 100 images and copy them into a working directory. The data contains faces of people 'in the wild', taken with different light settings and rotation. They appear to have been centered in this data set, though this need not be the case. There are a number of pre-processing steps we might wish to carry out before using this in any Deep Learning project. The paragraphs below list some of the most common.

Uniform aspect ratio: One of the first steps is to ensure that the images have the same size and aspect ratio. Most of the neural network models assume a square shape input image, which means that each image needs to be checked if it is a square or not, and cropped appropriately. Cropping can be done to select a square part of the image, as shown. While cropping, we usually care about the part in the center.

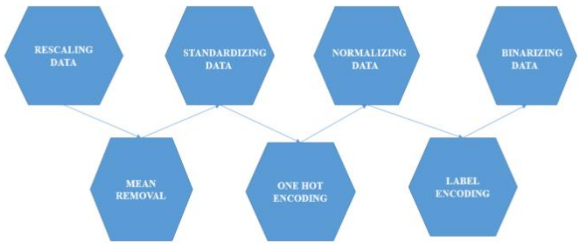


Fig3: Dataset Pre-processing

3.1.6 FEATURE EXTRACTION WITH HDF5 DATASET GENERATOR

HDF5 is binary data format created by the HDF5 group to store gigantic numerical datasets on disk (far too large to store in memory) while facilitating easy access and computation on the rows of the datasets. Data in HDF5 is stored hierarchically, similar to how a file system stores data. The feature extraction includes the following steps

1. We input an image to the network.
2. The image forward propagates through the network.
3. We obtain the final classification probabilities from the end of the network.

However, there is no "rule" that says we must allow the image to forward propagate through the entire network. Instead, we can stop the propagation at an arbitrary layer, such as an activation or pooling layer, extract the values from the network at this time, and then use them as feature vectors.

International Journal of Advanced Development in Science and Technology
IJADST Volume :3 Issue : 03 April 2021 www.ijadst.com e-ISSN: 2582-1059

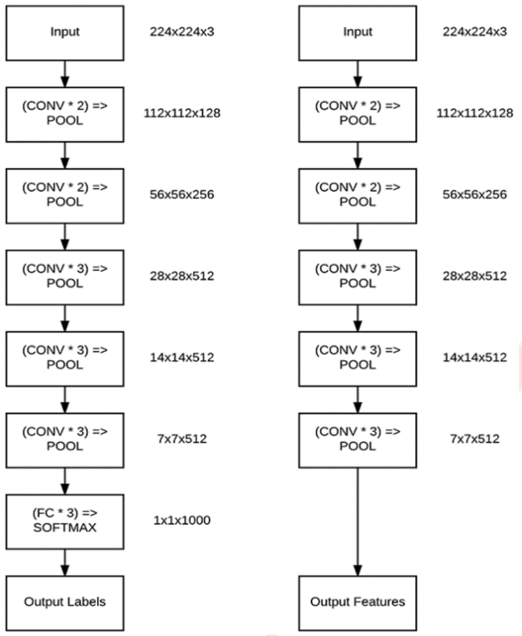


Fig4: Feature Extraction

In the figure above, Left: The original Resnet network architecture that outputs probabilities for each of the 1,000 ImageNet class labels. Right: Removing the FC layers from VGG16 and instead returning the output of the final POOL layer. This output will serve as our extracted features. Along with the layers in the network, we have also included the input and output shapes of the volumes for each layer. When treating networks as a feature extractor, we essentially "chop off" the network at an arbitrary point (normally prior to the fully-connected layers, but it really depends on your particular dataset). Now the last layer in our network is a max pooling layer (Figure 4.6, right) which will have the output shape of 7×7×512 implying there are 512 filters each of size 7×7. If we were to forward propagate an image through this network with its FC head removed, we would be left with 512, 7×7 activations that have either activated or not based on the image contents.

Therefore, we can actually take these 7×7×512 = 25,088 values and treat them as a feature vector that quantifies the contents of an image. If we repeat this process for an entire dataset of images (including datasets that resnet was not trained on), we'll be left with a design matrix of N images, each with 25,088 columns used to quantify their contents (i.e., feature vectors). Given our feature vectors, we can train an off-the-shelf machine learning model such a Linear SVM, Logistic Regression classifier, or Random Forest on top of these features to obtain a classifier that recognizes new classes of images.

Keep in mind that the CNN itself is not capable of recognizing these new classes instead, we are using the CNN as an intermediary feature extractor.

The downstream machine learning classifier will take care of learning the underlying patterns of the features extracted from the CNN. Later in this chapter, I'll be demonstrating how you can use pre-trained CNNs (specifically resnet) and the Keras library to obtain >95% classification accuracy on image datasets such as Animals, CALTECH-101, and Flowers-17.

Neither of these datasets contain images that resnet was trained on, but by applying transfer learning, we are able to build super accurate image classifiers with little effort. The trick is extracting these features and storing them in an efficient manner.

3.1.7 DATABASE INTEGRATION

A database is a collection of one or more related tables of data stored in rows and columns. By this definition even a spreadsheet is a simple database. However, many business databases consist of multiple files that are interrelated. Databases may be searched, sorted, and summarized to display information. Most businesses would not function without databases of information. For example, imagine a bank trying to function without its database of customer and account information. Information is usually entered into a database using a form. Each form normally corresponds to a single record or row in the database. Each field in the form normally corresponds to a single column or cell in that record. The form helps ensure that correct information is entered. Drop down menus in particular help ensure that users do not type in gibberish.

DATA CONSOLIDATION

Data consolidation physically brings data together from several separate systems, creating a version of the consolidated data in one data store. Often the goal of data consolidation is to reduce the number of data storage locations. Extract, transform, and load (ETL) technology supports data consolidation.

ETL pulls data from sources, transforms it into an understandable format, and then transfers it to another database or data warehouse. The ETL process cleans, filters, and transforms data, and then applies business rules before data populates the new source.

DATA PROPAGATION

Data propagation is the use of applications to copy data from one location to another. It is event-driven and can be done synchronously or asynchronously. Most synchronous data propagation supports a two-way data exchange between the source and the target. Enterprise application integration (EAI) and enterprise data replication (EDR) technologies support data propagation.

DATA VIRTUALIZATION

Virtualization uses an interface to provide a near real-time, unified view of data from disparate sources with different data models. Data can be viewed in one location, but is not stored in that single location. Data virtualization retrieves and interprets data, but does not require uniform formatting or a single point of access.

DATA FEDERATION

Federation is technically a form of data virtualization. It uses a virtual database and creates a common data model for heterogeneous data from different systems. Data is brought together and viewable from a single point of access. Enterprise information integration (EII) is a technology that supports data federation. It uses data abstraction to provide a unified view of data from different sources. That data can then be presented or analyzed in new ways through applications. Virtualization and federation are good workarounds for situations where data consolidation is cost prohibitive or would cause too many security and compliance issues.

DATA WAREHOUSING

Warehousing is included in this list because it is a commonly used term. However, its meaning is more generic than the other methods previously mentioned. Data warehouses are storage repositories for data. However, when the term "data warehousing," is used, it implies the cleansing, reformatting, and storage of data, which is basically data integration.

Integration of data from multiple, heterogeneous databases is a commonly encountered scenario in information retrieval systems where the user is to be provided with a unified view of information. These data sources may be from different vendors, may comprise different schemas and, could be physically at different locations.

3.1.7 MOBILE APP DEVELOPMENT

React Native is a framework that builds a hierarchy of UI components to build the JavaScript code. It has a set of components for both iOS and Android platforms to build a mobile application with a native look and feel. Mobile development has witnessed unprecedented growth.

According to statistics, mobile applications will generate an estimated 188 billion U.S. dollars in revenue via app stores, advertising and in-app purchases by the year 2020. Single and business users require high-standard apps with flawless performance, multiple screens, easy navigation and good design. On the other side, high-performing, good quality native apps are very time-consuming to develop compared to cross-platform apps that provide faster development but compromise on performance and support. React Native seems to be a viable solution for building high-quality apps in a short time with the same

IJADST
International Journal of Advanced Development in Science and Technology
Volume :3 Issue : 03 April 2021 www.ijadst.com e-ISSN: 2582-1059

performance and user-experience standards that native apps provide.

4. RESULTS AND DISCUSSION

Data preprocessing involves three steps namely data collection, data preprocessing and data augmentation. Various datasets were collected and one example among the collected dataset can be found below

Fig5: Dataset Collected

These datasets are then preprocessed to form an equal aspect ratio so that it can be made ready for training with the model. Now the model undergoes the process of feature extraction in which the features of the image are extracted and converted from string to vector form so that it can be trained by the proposed algorithm using the HDF5 dataset generator.

The blow figure shows the backend running of the feature extraction code with datasets given as input and getting the extracted feature in the hdf5 folder

After processing the hdf5 file gets generated which is then ready for undergoing the training process with the algorithms using network network surgery. In the below figure shows the file generated after hdf5 feature extraction in the output folder

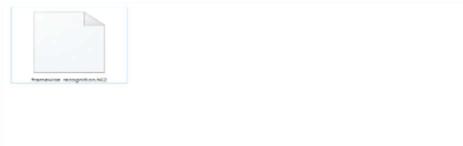


Fig6: Feature Extraction Output

Below screenshot shows that detection of walk and action recognition based on open pose.

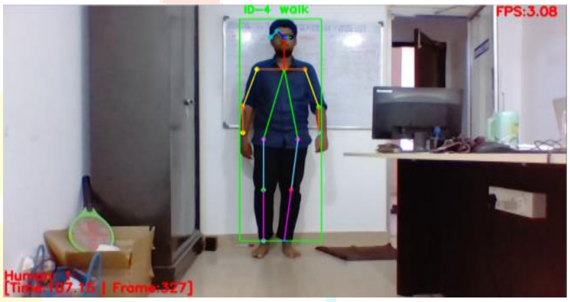
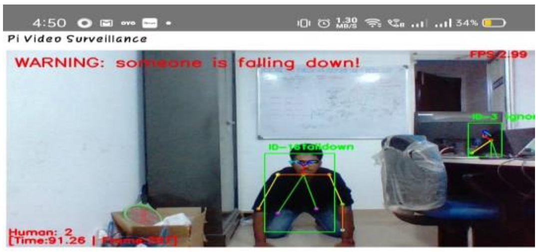


Fig7: Walk detection

Below screenshot shows the Human Action detection of falling down.

Fig8: Falling down detection

Above screenshot shows that an abnormal activity has been detected and as soon as it send the notification to the required person.



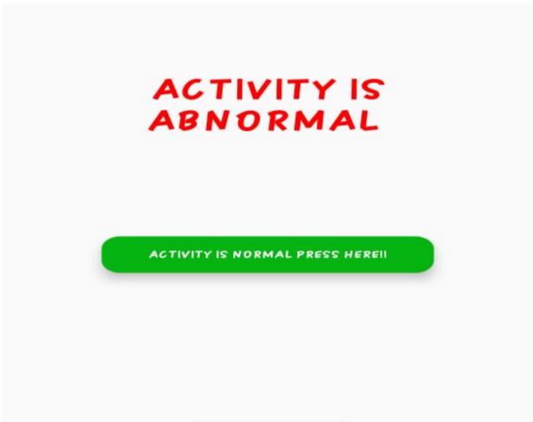


Fig9: Abnormal Activity Screen

Above screenshot shows that an abnormal activity has been detected

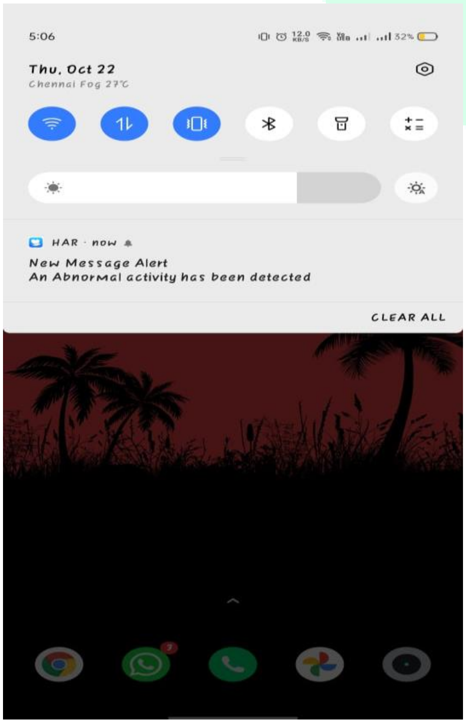


Fig10: Alert Notification

Thus, this project helps in effective human recognition which helps in identifying the human activities and intimates about abnormal condition that will help a care taker of an elderly person.

5. CONCLUSION

This project a human action recognition and human gesture recognition system which can automatically recognize the human daily activities using the currently prevailing deep learning approach. We develop an effective skeleton information based HAR. It will recognizethe human activities effectively.

We have developed a react native application so that we can view the live streaming with a notification when an abnormal activity is detected. So, we can save our grandparents from sudden health issues and can also help the old age homes for taking care of the elder people. Thus, this project helps in effective human recognition which helps in identifying the human activities and intimates about abnormal condition that will help a care taker of an elderly person. Thus, this project helps in effective human recognition which helps in identifying the human activities and intimates about abnormal condition that will help a care taker of an elderly person.

International Journal of Advanced Development in Science and Technology
Volume :3 Issue : 03 April 2021 www.ijadst.com e-ISSN: 2582-1059

6. REFERENCES

[1] Alfredo Nazabal, Student Member, IEEE, Pablo Garcua-Moreno, Antonio Artes-Rodruguez, Senior Member, IEEE, ZoubinGhahramani. “Human Activity Recognition by Combining a Small Number of Classifiers” [2017, Vol No 2716-6112]

[2] Anshul Mittal, Pradeep Kumar, ParthaPratim Roy, Raman Balasubramanian And Bidyut B. Chaudhuri “A Modified-Lstm Model For Continuous Sign Language Recognition Using Leap Motion” [2019, Vol No1558-1748]

[3] Cho NilarPhyo , Student Member, Ieee, ThiThi Zin, Member, Ieee And Pyke Tin “Deep Learning For Recognizing Human Activities Using Motions Of Skeletal Joints” [2018,Vol No: 0098-3063]

[4] Chhavi Dhiman, Member IEEE, Dinesh Kumar Vishwakarma, Senior Member, IEEE “View-invariant Deep Architecture for Human Action Recognition using Two-stream Motion and Shape Temporal Dynamics”[2020, Vol No:0098-3063]

[5] Dapeng Tao, LianwenJin, Member, Ieee, Yuan Yuan, Senior Member, Ieee, And Yang Xue “Ensemble Manifold Rank Preserving For Acceleration-Based Human Activity Recognition” [2016, Vol No: 2162-237X]

[6] Hao Zhang, Member, Ieee, Wenjun Zhou, Member, Ieee, And Lynne E. Parker, Fellow, Ieee “Fuzzy Temporal Segmentation And Probabilistic Recognition Of Continuous Human Daily Activities” [2015, Vol No: 2168-2291]