|  |  |
| --- | --- |
| A picture of a winding road and trees  Hand Gesture Recognition for SMART TV Control  A Deep Learning Project | Abstract  This Project involved building a 3D Convolutional Neural Network (CNN) to correctly recognize hand gestures by a user to control a smart TV. The gestures are continuously monitored by the webcam mounted on the TV.  By Saravanakumar PERUMAL and Saunak MALLIK  ML-C64 |

Project: Hand Gesture Recognition for SMART TV Control

**by Saravanakumar PERUMAL and Saunak MALLIK | ML-C64**

Contents

[Project overview 1](#_Toc184816738)

[General Information 1](#_Toc184816739)

[Project pipeline and Conclusions 2](#_Toc184816740)

[**CNN+RNN Architecture for Gesture Recognition** 2](#_Toc184816741)

[Business Conclusions 6](#_Toc184816742)

[Technologies Used 6](#_Toc184816743)

[Acknowledgements 6](#_Toc184816744)

# Project overview

**Description**: This Case study involved building a 3D Convolutional Neural Network (CNN) to correctly recognize hand gestures by a user to control a smart TV. The gestures are continuously monitored by the webcam mounted on the TV. Each gesture corresponds to a specific command.

Here, we have used 5 hand gestures as part of the scope of this project, namely Thumbs up (Increase the volume), Thumbs down: Decrease the volume, left swipe ('Jump' backwards 10 seconds), Right swipe ('Jump' forward 10 seconds), Stop (Pause the movie).

Note: This case study is for learning/ educational purposes only.

**Hashtags**. #Gesture Recognition #SMART TV controls #Hand Gestures automation #Hand Gesture automated controls #IOT

# General Information

* **Domain**. HOME Electronics (SMART TV) Industry.
* **Background**. HOME electronics companies manufacture state of the art SMART televisions. We have developed a cool feature in the smart-TV that can recognize five different gestures performed by the user which will help users control the TV without using a remote.
* **Dataset**. The training data consists of a few hundred videos categorized into one of the five classes. Each video (typically 2-3 seconds long) is divided into a sequence of 30 frames(images). These videos have been recorded by various people performing one of the five gestures in front of a webcam - like what the smart TV will use.
* **Dataset classes**. The data set contains the following 5 hand gestures as part of the scope -
  + Thumbs up: Increase the volume
  + Thumbs down: Decrease the volume
  + Left swipe: 'Jump' backwards 10 seconds
  + Right swipe: 'Jump' forward 10 seconds
  + Stop: Pause the movie
* **Outcome of the model**. We have built a multiclass classification model using custom Convolutional Neural Network (CNN) in TensorFlow, which can accurately detect Hand gestures by a user to control a smart TV. The gestures are continuously monitored by the webcam mounted on the TV. Each gesture corresponds to a specific command.

Note: We have used one custom-built 3D CNN-model and another custom-built model using Transfer Learning from ResNET50.

# Project pipeline and Conclusions

* **Data Reading/Data Understanding**: Defined the path for train and test images
* **Model Building & training**: Create a custom 3D-CNN model, which can accurately detect 5 classes of hand gestures in the dataset. While building the model, rescaled the images to normalize pixel values between (0,1).
* **Optimiser and LOSS Function**. I have chosen **ADAM** optimiser and **Sparse Categorical Cross entropy** loss function for model training. I have trained the model for **50 epochs**.
* At each step, I have built a CNN model, compiled it, summarised it and then trained it. I have used **RELU** activation function and **SOFTMAX (because of 5 classes - multi-class classification problem)** in the last layer.
* **MODEL BUILDING**.

# **CNN+RNN Architecture for Gesture Recognition**

|  |  |  |  |
| --- | --- | --- | --- |
| **Experiment Number** | **Model** | **Result** | **Decision + Explanation** |
| **1) imread, imresize error** | CNN+RNN | **Throws scipy.misc import imread, imresize not supported** | **Used import cv2, cv2.imread cv2.resize to resolve the error.** |
| **2) Generators** | CNN+RNN | **Batch size and Number of batches are not matched** | **Try to improve the generator code to account the custom batches and try to add support to process left out end of batch data in training and test data to consume entire training and test data.** |
| **3) Image data & use of Pre trained Model** | CNN+RNN | **Accuracy: 0.19** | **Try to Included the all 30 images in the data folder.**  **Try to Used ResNet50 pre trained model Add added few custom Conv2D and max pooling layers** |
| **4) Allow few layers of ResNet** | CNN+RNN | **Accuracy: 0.43** | **Try to allow last few layers in ResNet50 and improve the accuracy**  **Add few LSTM layers** |
| **5) Final Model** | CNN+RNN | **Accuracy improved to 0.49** | **Training accuracy and test accuracy improving each epoc. As we have to stop on 50 epocs. We have got 49 % accuracy. We can still improve the model by running more epocs and more testing and training data. As the training data set is very small the accuracy is only moderate for the CNN+RNN model**  **To improve the accuracy we need more training data sets.** |

**3D - Architecture for Gesture Recognition**

|  |  |  |  |
| --- | --- | --- | --- |
| **Experiment Number** | **Model** | **Result** | **Decision + Explanation** |
| **1) Generators** | **3D CNN** | **Test Generator** | **Try to validate test and training data generation code** |
| **2) Bridge between 3DCNN and NN layer** | **3D CNN** | **ValueError** | **The Dense layer expects an input with shape (None, 115200) (which means a batch of 115200 features per example).**  **Try to use GlobalAveragePooling3D() to resolve the bridge between 3D CNN and NN layer** |
| **3) Resource Error** | **3D CNN** | **ResourceExhaustedError** | **Try to Reduce the batch size from 50 to 40** |
| **4) Build base 3D CNN Model** | **3D CNN** | **Accuracy: 0.55** | **Base model is built with 55% accuracy**  **Try to change dropout values and try to add more layers** |
| **5) 3D CNN Model** | **3D CNN** | **Accuracy: 0.64** | **Try to add more Convolutional and Max pooling layers.** |
| **6) Final Model** | **3D CNN** | **Accuracy: 0.80** | **Training accuracy and test accuracy improving each epoc. As we have to stop on 50 epocs. We have got 80 % accuracy. We can still improve the model by running more epocs and more testing and training data. As the training data set is very small, we have achieved 80% accuracy. To improve the accuracy further we need more training data sets.** |

# Business Conclusions

The multiclass classification model using a custom Convolutional Neural Network (CNN) in TensorFlow, can accurately detect hand gestures to control SMART TV features based on the hand gestures of Users. This would eliminate the need for Remote controls.

# Technologies Used

* python 3 with Jupyter Note Book
* libraries used - sklearn, statsmodels.api, numpy, pandas, warnings
* libraries used for data visualization - matplotlib, seaborn
* Tensorflow and Keras libraries for CNN Model building

# Acknowledgements

* This project and the data was provided by [IIITB] (<https://www.iiitb.ac.in/>) & [upGrad](https://www.upgrad.com/) learning platform.
* Please note that the dataset is for learning purposes only.

**Contact**

* Created by Saravanakumar PERUMAL ([saravana2u@gmail.com](mailto:saravana2u@gmail.com) |<https://github.com/Saravana2u/>) AND
* Saunak MALLIK ([saunak.mallik@outlook.com](mailto:saunak.mallik@outlook.com) |[https://github.com/saunakmallik2502/)](https://github.com/saunakmallik2502/)-)

- feel free to contact us!

* Project work repository: [Hand Gesture recognition](https://github.com/Saravana2u))