Minor Project- Report

Aug-2019-2020

**Course Guide**: Prof. Poornima A B

**Course Name & code**: Machine Learning(17CS6DCMPR)

**Semester:**    6                                                     **Date:** 21-05-2020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TITLE OF THE PROJECT | Flowers Image Classification using Convolutional Neural Networks and Unity 3D. | | | |
|  |  | | | |
| STUDENT NAME | Manoj V Khatokar | M Hemanth Kumar | Nithin S J | Manoj N |
| USN | 1DS17CS063 | 1DS17CS059 | 1DS17CS074 | 1DS17CS062 |
| INDIVIDUAL  CONTRIBUTION | Designing the front end using Unity 3D and configuring the database. | Building the Machine Learning Model. | Integrating front-end with back-end. | Designing the back-end and retrieval of image from the database using Flask API. |
| GUIDE | Prof. Poornima A B   Prof. Dr. Krishnan R | | | |
|  |  | | | |
| PROJECT ABSTRACT : | Automatic identification and recognition of Flower species in environments such as forests,mountains and dense regions is necessary to know about their existence. In recent years, plant species recognition is carried out based on the shape, geometry and texture of various plant parts such as leaves, stem, flowers etc. Flower based plant species identification systems are widely used. While modern search engines provide methods to visually search for a query image that contains a flower, it lacks in robustness because of the intra-class  variation among millions of flower species around the world. A software called Unity 3D was used to create an application that allows users to capture images of flowers and send it to the ML model for classification.  Hence, a Deep learning approach using Convolutional Neural Networks (CNN) is used to recognize flower species with high accuracy. Images of the plant species are acquired using the built-in camera module of a laptop/PC. | | | |
| PLATFORM USED  (H/W & S/w tools   used | Softwares : Unity 3D, Anaconda-Spyder, XAMPP   Languages : C#, Python, SQL, PHP.   Code Editors: Microsoft Visual Studio | | | |
|  |  | | | |
| Introduction | Plant species recognition based on flower  identification remain a challenge in Image processing and  Computer Vision community mainly because of their vast  existence, complex structure and unpredictable variety of  classes in nature. Because of these natural complexities, it is  highly undesirable to perform normal segmentation or feature  extraction or combining shape, texture and color features  which results in moderate accuracy on benchmark datasets.  Although some feature extraction techniques combining  global and local feature descriptors reaches state of the art  accuracy in classifying flowers, still there is a need for a  robust and efficient system to automatically identify and  recognize flower species at a larger scale in complex environments.  In Deep Learning, CNNs are specifically applied for Computer Vision applications that involve Image Classification and Object Recognition. Flower Species Recognition is a combination of both Object Recognition and Image Classification, as the system must detect a flower in the image as well as recognize which species it belongs to. To  recognize the flower species, an intelligent system must be  trained with larger set of images, so that it could predict the  flower species from its learned patterns. This approach is  termed as “Supervised Learning” which requires an external  dataset of images with labels to predict the label of an unseen  image. This research work uses Convolutional Neural  Networks (CNN) along with Transfer Learning as the  intelligent algorithm to efficiently recognize flower species in  real-time.  In Image Classifier a pretrained model was used. It was loaded as a pre-trained network, based on which we defined a new, untrained feed-forward network as a classifier, using ReLU activations and dropout.We trained the classifier layers using backpropagation using the pre-trained network to get the features. The loss and accuracy on the validation set were tracked to determine the best hyperparameters. | | | |
| List of papers/URLS referred | 1)<https://www.analyticsvidhya.com/blog/2020/02/learn-image-classification-cnn-convolutional-neural-networks-3-datasets/>  2)<https://www.geeksforgeeks.org/image-classifier-using-cnn/>  3)<https://www.mathworks.com/help/deeplearning/ug/transfer-learning-using-alexnet.html> | | | |
|  |  | | | |
| Design  {SYSTEM DESIGN DIAGRAM} | **CORE ARCHITECTURE**https://lh4.googleusercontent.com/dFXnljQn4tqzpIx1Ytf4BwiAL2FViOzouDr4wdb9Zy2iGFQRAv7Cm525vqn9WHv7BlZETPUq1ULmeUTrqhuWrW30YLhdXg1rObpi47t5l27R08SCPM8Waq5fTG80Bj1LS22eMXK8    **THE PROPOSED SYSTEM**  https://lh3.googleusercontent.com/CTtg3v2YQX0T64u93gzbmwFQOWR-2qCOwn4RISQc-F7dq4QlhrwawMrsSN1YH4Af1g1K-CT7dcYVYLaiyn108sVvSKNp99xnUQLzKqlFTQzYmTFFcecfl10Y_tsWOVWWRVk9m4qn | | | |
| Project Source Code Link (Github/ Google DRive) | <https://github.com/manojvkhatokar/Image_classifier_for_unity>  <https://github.com/manojvkhatokar/Object_Detection_Unity3D>  <https://github.com/manojvkhatokar/imgcls_utils> | | | |
|  |  | | | |
| Conclusion /FUTURE ENHANCEMENT | The proposed project serves to classify a given image against 102 classes of flowers using CNN and Web APIs.  In the current situation there is a lack of robustness because of the intra-class variation among millions of flower species around the world.  The underlying model can be also used for object detection as well by changing and training on the required datasets.  Therefore this piece of work can be leveraged and be used to classify more and more species of flowers and hence getting a better accuracy in the prediction. | | | |
|  |  | | | |
| Ui screenshots  of SAMPLE RESULTS | **UI DESIGN AND INPUT**  **https://lh3.googleusercontent.com/JzkUbSODSJ1K28dV4MmJcydzo5PWxPYwh7fVYhF6l_AQPmmXtXr7sOrET6mxM5arf1g8KvRVKqLUMcwsOm-YXpXUXGSXiY2pk6tnSgZBGTxGpV6UvKgCOoT_mb1LGnNDmsgjFZZW**  **OUTPUT**https://lh6.googleusercontent.com/Q8smLRTAKsXP5iSIi675MeacAlV2qZZ5BPklb0ItNlsC_SRWDR0Ip5FpNNB0QISgXf74a1uHLVu_sc4Fkm9axsFy2S24LbO4vG4bjTOkSdrfWJLPoWTd7qFhEBTfrU5jLGPlmuqK | | | |
| REFERENCES | 1. <https://www.sciencedirect.com/topics/computer-science/image-classification> 2. [https://www.tensorflow.org/tutorials/images/classification](https://www.tensorflow.org/tutorials/images/classificationhttps:/ieeexplore.ieee.org/document/8932908) 3. [https://ieeexplore.ieee.org/document/8932908](https://www.tensorflow.org/tutorials/images/classificationhttps:/ieeexplore.ieee.org/document/8932908) 4. <https://towardsdatascience.com/wtf-is-image-classification-8e78a8235acb> 5. <https://www.freecodecamp.org/news/how-to-build-the-best-image-classifier-3c72010b3d55/amp> 6. <https://katba-caroline.com/what-flower-is-this-developing-an-image-classifier-with-deep-learning-using-pytorch> | | | |