**Image Classification using Deep learning with Tensor Flow**

We as a group found an image classification dataset on Kaggle and used the dataset for training the model to predict the type of flower. We used Jupiter notebook platform for classification and training the model.

## [Deep learning](https://www.sciencedirect.com/topics/engineering/deep-learning) methods for image analysis are attracting increasing interest for application in a wide range of different research fields ([**Energy and AI**](https://www.sciencedirect.com/journal/energy-and-ai)

[Volume 15](https://www.sciencedirect.com/journal/energy-and-ai/vol/15/suppl/C), January 2024, 100330)

**GitHub repository link:**

https://github.com/Sai3Srujana/SJ\_AI.git

**Data selection and preprocessing:**

The data we took was online from Kaggle and uploaded the URL on to the Jupiter notebook to read the file as well as preprocess the data that can be used for training the model. Also, we compiled the model to get the precision, accuracy etc.

Link for dataset:

<https://www.kaggle.com/code/ryanholbrook/create-your-first-submission>

A screenshot of a computer

Description automatically generated

Here, we imported all the required libraries on to Jupiter notebook including tensor flow and Keras. After that we gave the storage repository path for it to read the file. We have total of 3670 images.

**Model selection and implementation:**

We need to consider following factors when selection the data for image classification:

* Characteristics of dataset: The dataset size, complexity, and image type.
* Complexity of model: The model should have the ability to capture all the retails and reduce overfitting.

For implementation, Tensor Keras API is used, and it simplifies image creation and training of the deep learning models.

**Defining the Model:**

A screenshot of a computer program

Description automatically generated

**Training the model:**

A screenshot of a computer

Description automatically generated

A computer screen shot of a computer code

Description automatically generated

A screenshot of a computer

Description automatically generated

**Compiling the model:**

A screen shot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Evaluation of performance metrics:**

After training the model, we need to evaluate the metrics, commonly we use accuracy, precision, Fi score, recall and confusion metrics.

* **Accuracy:** This explains overall correctness of the model.
* **Precision:** This explains the number of predicted positive instances that are positive.
* **F1 Score:** this is used to balance the precision and recall value.
* **Recall:** this indicates the number of actual positives in the model that are correctly predicted.
* **Confusion Metrics:** Representation of predicted and actual values.

Evaluating the model with different metrics gives us a detailed understanding of the model performance.

**Reference:**

## **[Energy and AI](https://www.sciencedirect.com/journal/energy-and-ai" \o "Go to Energy and AI on ScienceDirect)**

## [Volume 15](https://www.sciencedirect.com/journal/energy-and-ai/vol/15/suppl/C), January 2024, 100330

## **Current trends on the use of deep learning methods for image analysis in energy applications -** [**https://www.sciencedirect.com/science/article/pii/S2666546823001027**](https://www.sciencedirect.com/science/article/pii/S2666546823001027)

1. **RYAN HOLBROOK -** Create Your First Submission

<https://www.kaggle.com/code/ryanholbrook/create-your-first-submission>

1. **GitHub repository link -** https://github.com/Sai3Srujana/SJ\_AI.git