Identify cloud patterns by both cloud images and meteorological parameters using hybrid deep learning model

MSc Research Project
MSc Data Analytics

Indranil Bera
Student ID: 20144075

School of Computing
National College of Ireland

Supervisor: Vladimir Milosavljevic

National College of Ireland

Project Submission Sheet – 2021/2022

Student Name:	Indranil Bera
Student ID:	20144075
Programme:	MSc in Data Analytics **Year:** 2021-2022
Module:	MSc Research Project
Lecturer:	Vladimir Milosavljevic
Submission Due Date:	18/09/2022
Project Title:	Identify cloud patterns by both cloud images and meteorological parameters using hybrid deep learning model
Word Count:	8225

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the references section. Students are encouraged to use the Harvard Referencing Standard supplied by the Library. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action. Students may be required to undergo a viva (oral examination) if there is suspicion about the validity of their submitted work.

Signature:	Indranil Bera
Date:	18/09/2022

PLEASE READ THE FOLLOWING INSTRUCTIONS:

1. Please attach a completed copy of this sheet to each project (including multiple copies).
2. Projects should be submitted to your Programme Coordinator.
3. **You must ensure that you retain a HARD COPY of ALL projects**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. Please do not bind projects or place in covers unless specifically requested.
4. You must ensure that all projects are submitted to your Programme Coordinator on or before the required submission date. **Late submissions will incur penalties.**
5. All projects must be submitted and passed in order to successfully complete the year. **Any project/assignment not submitted will be marked as a fail.**

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Identify cloud patterns by both cloud images and meteorological parameters using hybrid deep learning model

Indranil Bera
x20144075@student.ncirl.ie

Abstract

The atmosphere is a complex interaction of different meteorological features. Weather prediction is one of the most prominent concerns in meteorological research. Weather forecasting can be very helpful to take preventive measures for any upcoming concerns, mitigate financial risk, life loss etc. The traditional way of forecasting is by Numerical weather prediction techniques which solve several complex differential equations related to atmospheric energy. NWP is very time-consuming and resource extensive. Deep learning algorithms become a popular way of dealing with huge weather data captured from weather stations. Cloud patterns are closely related to the weather parameters of that particular location. Several research was done on cloud pattern detection by cloud images from satellites, from ground-based camera but very less study was found by combining the cloud images and weather features. Here hybrid novel multi-modal architecture was developed using ground-based cloud images of Singapore and 9 weather features. Multi-Layer Perceptron (MLP) was implemented for numerical data and CNN-based 'MobileNet' pre-trained model was employed to extract features from the images. This hybrid model outperformed the previously built CNN-based pre-trained 'VGG16' model on the same dataset by 8%. Incorporating weather parameters in the image classification model can improve cloud classification significantly.

Keywords: Metrology, Forecast, Numerical Weather Prediction (NWP), Deep Learning, CNN, Multi-Modal, Multi-Layer Perceptron (MLP), VGG16, MobileNet

1. Introduction

Cloud is a visible mass of water drops formed by condensed vapour from the atmosphere. Clouds play an important part in weather and climate system. Cloud formation and patterns vary according to location depending on various meteorological parameters. Cloud patterns are very important to recognize to carry out different weather research in this community. The most efficient way is to take satellite images and detect them by meteorologists by performing experiments to understand the patterns of cloud images. But for large amounts of images, it is a very time-consuming and error-prone way. So there is a constant need to automate this cloud classification task in recent years. Weather forecasting is complex and often has numerous factors associated with meteorological features of the atmosphere. There are multiple approaches available to understand weather behaviour by analyzing the cloud patterns. Numerical weather prediction is a traditional way of forecasting by solving critical differential equations which is resource intensive and very time-consuming process (Pielke, 2002). Machine learning and deep learning algorithms become popular in weather prediction after the availability of a large amount of data. The weather data collection is done at different atmospheric levels with modern instruments. Now with the help of global weather stations and satellite technologies weather data is even available in real-time. Deep learning algorithms can help to identify patterns in data and images. Cloud formation is a localized phenomenon so patterns vary according to the location. So it is not possible to build one model which can identify clouds from all over the world (Pathan et al., 2019). Image-based solutions became popular after the introduction of CNN-based algorithms for image classification. In this study, cloud images and weather data were used to develop a multi-modal hybrid deep learning model that can classify cloud patterns. Images taken were captured from a ground-based camera system in Singapore with five cloud types. Singapore is a country located in Southeast Asia which contributes a lot to weather research and the development of open-source weather datasets for research purposes.

1.1 Motivation

Cloud is an important part of the weather and identifying cloud patterns is a vital part of meteorological research. The traditional way of cloud pattern identification is time-consuming, error-prone, resource-burning, costly and sometimes biased by human judgements. There is a constant need for automation for this job. Cloud classification can help forecast weather forecasting, solar energy and wind energy generation prediction, rainfall, etc. These can further lead to many more advantages in the socio-economic life of human beings. This research can help to save electricity, make better decisions in agriculture, disaster management, aviation, and ship navigation, minimize economic loss and save human lives. This research concentrates on the building of multi-modal algorithms based on ground-based images and weather parameters. Very little research was done in the weather prediction domain using this kind of algorithm. So this research is going to contribute to the Data Science community, especially in the meteorology domain.

1.2 Research Question

How do ground-based cloud images and meteorological parameters identify cloud patterns in a small area of Singapore?

The main objectives of this research are –

- Build a MLP model by numerical weather data to classify cloud patterns
- Build a CNN based transfer learning algorithm for image classification model to predict cloud types
- Build a multi-model deep neural network which will be able to identify cloud types by extracting features from the ground-based images and numerical weather parameters. This study will contribute to weather prediction and meteorological science.

2. Literature Review

Cloud has an enormous role in weather forecast and the climate system of Earth. In recent years lots of research have been conducted on cloud formation and identifying cloud patterns. Each cloud type has a distinct radiative effect so for meteorologists it is incredibly significant to identify the cloud patterns correctly. Cloud formation depends on the weather parameters of a particular location. It is not an easy task to understand cloud formation or predict the weather as these natural phenomena are controlled by tons of complex factors related to the atmosphere. There are multiple approaches available to understand weather behaviour by analyzing the cloud patterns. The most traditional way is numerical weather prediction (NWP) which required a large amount of weather data to solve heavy mathematical equations. The modern weather data collection process uses doppler radars, weather balloons, satellites, radiosondes, buoys, and other instruments in different atmospheric levels (Mahajan and Fataniya, 2019). Plenty of weather stations are collected those data and available in real-time. The history of Machine learning and deep learning techniques in meteorology is more disruptive. It avoids laborious data handling but can forecast the weather by using data that has been recorded in the past. Machine learning and deep learning algorithms are capable of performing a wide range of complex tasks in this area by utilizing numerical data. Image-based solutions became popular after the introduction of advanced neural network algorithms for image classification. To explore cloud formation and define its nature, several researchers have adopted distinct methodologies, which are mentioned below and divided into distinct domains.

2.1 Traditional Numerical Weather Prediction (NWP) for Studying Cloud Formation

The foundation of current meteorology science is numerical weather prediction (NWP).
NWP solves partial differential equations relating to fundamental conservation laws such as momentum, mass, energy, and water vapour. To predict weather conditions, historical data is used, and initial conditions are given as current atmospheric parameters to solve PDEs. Historic data is utilized to forecast weather conditions, and initial conditions are given as current atmospheric parameters to solve PDEs. Multiple complicated mathematical models must be equated in this methodology. From 1920 through 1950, tremendous progress was made, and in 1950, Charney et al. developed the first functional NWP model (1950). NWP became popular after the invention of

powerful computer systems in the 1960s because it can solve equations more quickly and accurately. During this period different experiments were performed as operational practice in the domain of dynamic meteorology, advanced numerical analysis, atmospheric observations etc. using electronic computers. But the different partial differential equations are very sensitive to the given initial conditions and most of the time it not possible to provide initial conditions correctly as it is not always possible to understand the present behaviour of the atmosphere at a particular location (Pielke, 2002). Meteorological parameters vary according to the location of the observations. That's why each country has their own weather data collection instruments and strategic deployment of the weather stations. For example, Met Éireann is part of the HiRLAM (High-Resolution Limited Area Model) group which covers most major countries of Europe and 10 European weather services. NWP contributed significantly in the last two decades to identify cloud cover and types to investigate solar energy forecasting. Mittermaier, M. (2012) critically assessed the surface cloud formation and methods to verify cloud forecasts. The researcher developed four Met Office Unified Models (MetUM) based on cloud amount and height from the location grid to compare manual and automated observations. The results showed compromise in observations while tuning the physics of the model. HARMONIE model and ground-based synoptic stations (SYNOP) are the most advanced NWP models available to date (Kurzrock et al., 2018).

Though numerical forecasting models are the foundation of modern weather forecasting, their implementation is complex and time demanding. Setting up a method for collecting data and powerful computational systems requires considerable interest and massive investment. With this method, small-scale projects are extremely challenging to achieve.

2.2 Machine Learning and Deep Learning based Weather Prediction by using Meteorological Data

Machine Learning (ML) is a subset of Artificial Intelligence (AI) that uses computer programming to create techniques that are backed by mathematical and statistical algorithms. The models can learn or recognize patterns in historical data and predict output parameters. AI's history has been more disruptive than NWP's. McCulloch and Pitts developed the first neural network in 1943, but the algorithms were unstable and difficult to apply to the available datasets After the discovery of the backpropagation (Rumelhart, Hinton and Williams, 1986) principle, ML and NN became accurate enough to employ in real-world datasets. One of the primary causes for the delay in practical implementation was a lack of data. After the boom of the world wide web and the invention of parallel processing by the Graphical Processing Unit (GPU), ML become popular after 2000s. Large datasets of weather data are utilized to understand cloud patterns and predict different weather variables. Initially, DL and ML were employed to fine-tune and optimize the outputs of NWP models for weather prediction. Three Multi-Layer Perceptron (MLP) models were used to forecast time series data of a particular weather station (Schizas et al.,2002).

Substantial research has been performed on this subject in recent years to investigate cloud patterns and formation in weather prediction of different areas. By utilizing machine learning methods in the remote sensing sector, a cloud detection and categorization method was developed (Tian, Chen and Liu, 2019). Accurate data from the atmosphere and the earth's surface are essential for these types of research models. The data used in this study was from CrIS which is the world's most advanced hyperspectral sounder[1]. Using the full spectral resolution improved the cloud detection index (FCDI). The brightness temperature (BT) from the community radiative transfer model was used to create the FCDI. Following the detection of cloud covers using BT simulations, cloud types were identified using machine and deep learning techniques. As a classification technique, Extreme Learning Machine (ELM), Support Vector Machine (SVM), and MLP were used. Among these, ELM outperformed the others. The ELM comprised three layers: input, hidden, and output. The hidden layer was constituted of 500 nodes with values assigned at random so that it could learn in a single step. The classification algorithm correctly classified clear and overcast skies with an accuracy of 80%. The researchers explained the lower accuracy to a shortage of labelled validation data.

The backbone of Deep learning algorithms is correctly labelled data. That's why to generate data different instruments were used. A cloud identification algorithm was developed by Poulsen et al. (2020) using data from Sea and Land Surface Temperature Radiometer (SLSTR). The algorithm was a unique Feed Forward Neural Net capable to identify cloud cover in the polar region. The results are compared with the existing SLSTR models based on Bayesian, Empirical and Probabilistic approaches to land and sea. The NN used 22 features as input which consist of 9 spectral channels, satellite, latitude, longitude, surface type flags, zenith angle ancillary information and solar zenith angle. Then in there were four hidden layers having 32 neurons in each layer. 'Leakyrelu' activation function was used and dropout was implemented with 0.80 probability. In the output layer 'softmax' was used and as an optimizer 'Adam' was used. Surprisingly sea level data had better accuracy than the surface. The average accuracy of the model was 93% and the NN algorithm was slightly better in Antarctica than in the Arctic.

Following a thorough analysis of the ML and DL algorithms used in this domain, it is observed that tremendous advancements have been made in weather prediction and cloud identification in recent years. With a large accurate dataset of meteorological features for a specific location, the algorithms perform admirably. Different image-based NN algorithms can also classify cloud cover and types using labelled image datasets, which are discussed in the next section.

2.3 Identify Cloud Patterns by CNN based algorithms using Cloud Images

CNN based deep learning algorithms can handle images and extract patterns or features from images. Human eyes can identify different images and can classify basis on their structure, colour, shape, form etc. In the same manner, neural network algorithms work to classify the images. Several research is carried out to classify cloud images by utilizing majorly satellite images and ground-based images. After the invention of efficient optimizers image classification algorithms become more accurate and got ready for practical implementation (LeCun, Yoshua, and Hinton, 2015). With increased computation power and availability of large image datasets CNN based image classification algorithms become popular in weather research.

There are very few publicly available cloud image dataset available for research which has different cloud types annotated by expert meteorologists. One of the prominent ground based datasets is International Cloud Atlas published by the World Meteorological Organization (WMO) Sinko et al. (2019) developed a CNN based AlexNet algorithms using this dataset to classify four types of cloud patterns namely Altocumulus, Cirrus, Cumulonimbus, and Cumulus. The original dataset had 10 types of clouds but only 4 are taken for the study to reduce false alarms due to class

[1] www.l3harris.com/all-capabilities/cross-track-infrared-sounder-cris

imbalance. The images are resized into 227×227 pixels to match the input of AlexNet architecture. It is a forward network having multiple hidden layers and each layer has a functional feature map which extracts features with multiple neurons. The model had eight layers with learnable parameters. ‘Relu’ was used as an activation function and three Max Pool layers of 3×3 filter are used throughout the model building. As this was a multi-class classification problem in the output layer ‘Softmax’ was used which gave the output of 4 cloud types. The model achieved 81% overall accuracy. Researchers concluded that if the dataset included more input images of each class and used high dynamic range (HDR) images, the accuracy may have been improved.

Another research was done on ground-based cloud image classification by Rhee and Phung (2018). The SWIMCAT dataset had 784 ground-based cloud images captured from a Singapore technology university camera system. It was consisting of 5 types of clouds with 125×125 resolution. As the dataset had small numbers of images that’s why image augmentation techniques were used to increase the training dataset. Dropout layers were used to handle the overfitting of the model and ‘RMSprop’ optimizer was used during the building of the CNN architecture. The model achieved an accuracy of 83% after 1000 epochs and 5 k-cross validations. To achieve this accuracy model training was computational heavy and consumed a lot of time. But this shows the capability of CNN based architecture to identify cloud patterns even in a small dataset. A novel approach was taken by the researchers (Dey Roy et al., 2021) on the same ground-based SWIMCAT dataset to classify the cloud images by pretrained CNN based models. Cloud images had five types namely Clear Sky, Patterned clouds, Thick dark clouds, Thick white clouds, and Veil clouds. Classification algorithms were implemented in two stages. In the first stage, cloud images were classified into two groups namely clear and non-clear. Then in the second stage, non-clear clouds were classified into other four types of clouds. CNN-based pre-trained models were used for both classification purposes. Publicly available Vgg16 and MobileNet pre-trained models were used for this study. The final solution achieved 84.5% accuracy. AlexNet, GoogleNet, RestNet 101 were also used but did not perform well. Researchers noticed that number of epochs and data augmentation played a vital role in the classification accuracy.

It is observed from the above review that cloud image classification is a popular domain for study and getting attention from researchers. CNN based architecture is well established now for image classification. The publicly available pre-trained models can be very useful for small datasets having less no of images belonging to each class. The next part of the literature review focuses on the hybrid deep learning model by using cloud images and numerical weather parameters.

2.4 Hybrid approach by analyzing cloud images and weather parameters

Hybrid models are built by combining different types of data as input and applying different deep learning models for each type of dataset. It is a relatively new concept in deep learning. There are limited resources available related to the weather domain which uses multimodal deep learning algorithms.

Yuan, Jiang, Li and Huang (2019) developed a hybrid deep learning model capable of handling mixed inputs of image data and numeric data. The main motto behind the research was to build a flexible architecture which can be applied to any domain and a variety of datasets for feature extraction purposes. Multiple input data went through the appropriate model for feature learning and then concatenated either by feature or channel. After that ensemble feature acts as an input to the target learning model. Categorical cross-entropy is used as a loss function. The researchers tested the architecture with numeric data with MLP and image data with pretrained CNN based models. The experiment gave good results and the accuracy increased by 4% than using a standalone model of any deep learning model.

The spectral ratio is critical in cloud image identification. By integrating ANN and Genetic algorithms, an innovative cloud classification model was proposed (Pallavi and Vaithiyanathan, 2019). The main goal was to improve the spectral property so that cloud images could be detected more efficiently. Images with band ratios were given into the model together with RGB values and spectral data as an ANN model. With crossover and mutation algorithms, a standard genetic algorithm was employed to change the weights of the neural network. This hybrid technique improved accuracy by 3% over the ANN model.

Another notable study was identified in Yokohama that used both cloud images and weather characteristics for weather prediction (Tsukahara, Fudeyasu and Fujimoto, 2020). The aim was to classify the weather into two categories: rainy and sunny. Humidity was used as a numerical weather parameter collected through a publicly available weather API. There were 28100 cloud images of 640×480 size. To process the image dataset, an advanced CNN-based deep learning architecture MobileNet pre-trained on the ImageNet dataset was applied. Another MLP model handled the numerical weather data. The final hybrid model had four hidden layers as well as softmax as an activation function. The model was predicted with 83.6 per cent accuracy. It achieved 10% higher accuracy than the ResNet model using only the image dataset. In most studies, incorporating weather parameters improved the outcomes. Combining weather variables with ground-based cloud images could therefore be a promising field and strategy to explore.

3. Research Methodology

To achieve the objectives of the project it must go through a step-by-step process. In this section, all the strategies and methodologies are discussed carefully. This research was carried out by Knowledge Discovery in Databases (KDD) methodology as this a multi-label classification problem where data mining plays a vital role. Three approaches were taken to solve the goals. A CNN based transfer learning model called VGG-16, was used only for cloud images. The second one was using Multilayer Perceptron (MLP) which is a popular feed-forward Artificial Neural Network (ANN) used for numerical weather data. In the last multi-modal or hybrid approach both numeric and image data were used to classify cloud images. Fig. 1 explains how the KDD methodology is incorporated into this research.

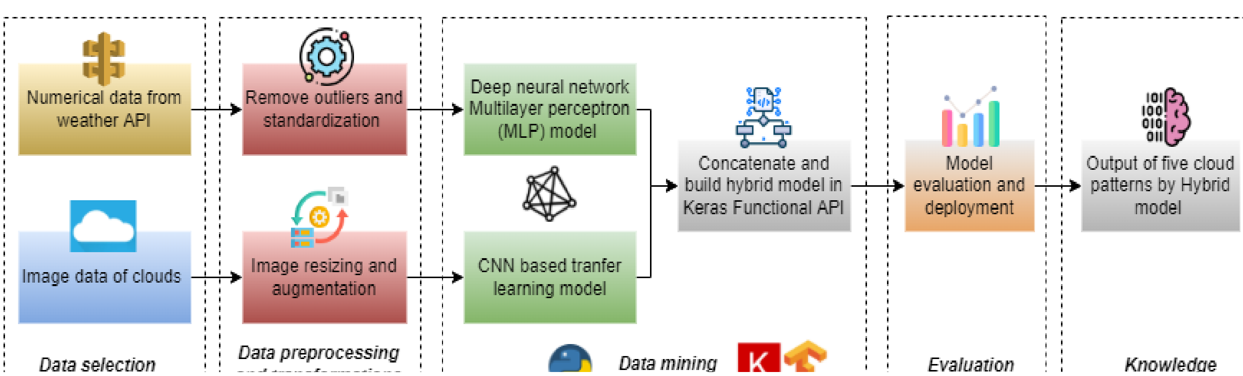

Fig. 1 KDD Process Flow implemented in the project

3.1 Data Selection and Dataset Description

The project had two types of datasets. One is ground based cloud image dataset and the other one is the numerical dataset of weather parameters. Details are discussed below –

a. *Ground based cloud image dataset – SWIMCAT*

SWIMCAT dataset is a ground-based cloud image dataset consisting of 784 images of 125×125 resolution (Dev et al., 2015). The dataset is small but has 5 types of clouds labelled by meteorologists of Nanyang Technological University in Singapore. Images were taken from ground based camera system located on the university campus from the year 2013 to 2014 for around 17 months.

Fig. 2 Five types of clouds in the SWIMCAT dataset
(a) Clear sky (b) Patterned clouds (c) Thick dark clouds (d) Thick white clouds (e) Veil clouds.

The cloud data had one CSV file for each folder type where the time and date of the captured images were given. Below are those details -

- Date - Date in YYYYMMDD format
- Hour - Hour in HH format
- Min - Minute in MM format
- Sec - Second in SS format

This dataset was released on a Creative Commons license[2] which allows using for non-commercial purposes.

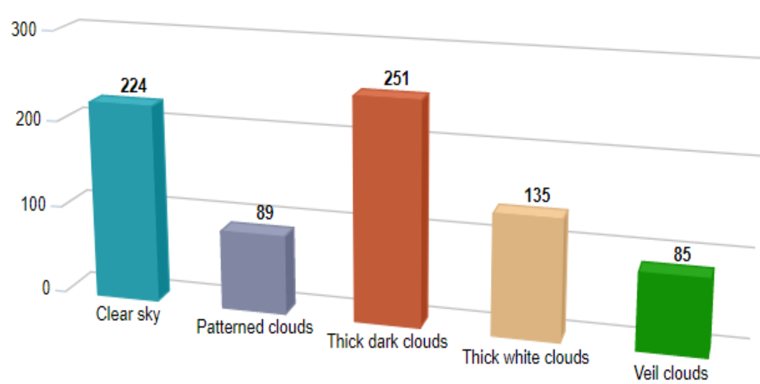

Fig. 3 Counts of five types of clouds in the SWIMCAT dataset

[2] https://creativecommons.org/licenses/by-nc/4.0/

b. Numerical weather parameter dataset created by open source weather API

With the help of time and date information of the captured images in SWIMCAT dataset, this numerical dataset was prepared using Visual Crossing weather API[3]. Co-ordinate of the location (1:34°N, 103:68°E), Date, Time was taken as inputs to this API which returned 11 weather parameters. The details of the parameters are mentioned below in table 1. All these variables were captured in a CSV file and stored locally for all 784 images of clouds.

Name	Description	Data Type
Temperature	Average temperature	Float
Dew Point	Dew Point	Float
Relative Humidity	Relative Humidity	Float
Heat Index	Heat Index	Int
Wind Speed	2 minute average of wind speed	Float
Wind Direction	2 minute average of wind direction	Float
Precipitation	Amount of liquid equivalent precipitation	Float
Visibility	Distance that can be viewed	Float
Cloud Cover	Percentage of sky that is covered by cloud	Float
Sea Level Pressure	Sea Level Pressure	Float
Weather Type	Weather types reported by weather station	String

Table 1 Description of weather dataset

3.2 Data Preprocessing and Transformations

As this study had two distinct datasets so both had different preprocessing and transformation techniques before feeding into the deep learning models.

3.2.1 Drop unnecessary columns in Weather Dataset

The data fetched from weather API had unnecessary columns which were not relevant to study. Wind Gust, Wind-Chill, Sea-Level Pressure, and Weather Type were those irrelevant variables which were dropped from the main dataset. The clean version of the weather dataset had eight columns which were used for further treatments.

3.2.2 Label Encoding for categorical target variable in both Datasets

The target variable was the cloud types which had five types of string values in the merged dataset. To apply deep learning algorithms it was necessary to convert those into multi-level integer variables. It was done by applying 'LabelEncoder' which is a popular function of 'sklearn' preprocessing library. Keeping in mind that it was a multi-class classification problem target cloud classes were converted into 0, 1, 2, 3, and 4.

3.2.3 Normalization of the numerical variables in Weather Dataset

A normalization is a standard approach that must be followed for deep learning with numerical data. The training and performance of algorithms are greatly influenced by data normalization

[3] https://www.visualcrossing.com/resources/documentation/weather-api

(Moeeni and Bonakdari, 2018). In this case, data columns were normalized by subtracting the mean (μ) from a column value and dividing the result by the column's standard deviation (σ). As a result, the numerical dataset was normalized and ready for model input.

3.2.4 Resize images according to the input of the model

Every image classification algorithm has a predefined input size for the network. Image size plays a vital role in image processing algorithms (Richter et al., 2021). Here VGG16 and Mobile Nets were used which had resolutions of 224x224 and 299x299 respectively. Resizing was done by the 'resize' function of 'OpenCV' library in Python programming language. The image sizes of the cloud image dataset had 125x125 resolution which was upsampled to the desired size of the above-mentioned input layer of the networks.

3.2.5 Create compressed NumPy array for images

Numpy is a well-established library of Python that helps in numerical computing. Compressed Numpy arrays were created as NPZ data files for all the numerical weather variables and image data. So the final NPZ data frame had information from both datasets which can be easily accessible.

4. Design Specification

The study contains three models which consumed two mentioned datasets earlier. All the design specifications are distinct to each model. Every algorithm has its own criteria of the input layer and preprocessing steps. The models used in this study are discussed briefly below.

4.1 Model 1 - Multi-layer perceptron (MLP)

The Multilayer Perceptron (MLP) is a deep learning method used to compute numerical features. It is a feed forward artificial neural network containing multiple numbers of neurons. MLP consists of three major layers – the input layer, hidden layers, and output layer. The input layer is exposed to input data which passes the input to the first hidden layer. Hidden layers can be multiple and each layer has several neurons. Weights and bias are updated in each hidden layer and propagated to the next layer. Backpropagation is used for the learning mechanism to adjust the weights and decrease the loss. As it's a feed forward network data flows from input to output via hidden layers,

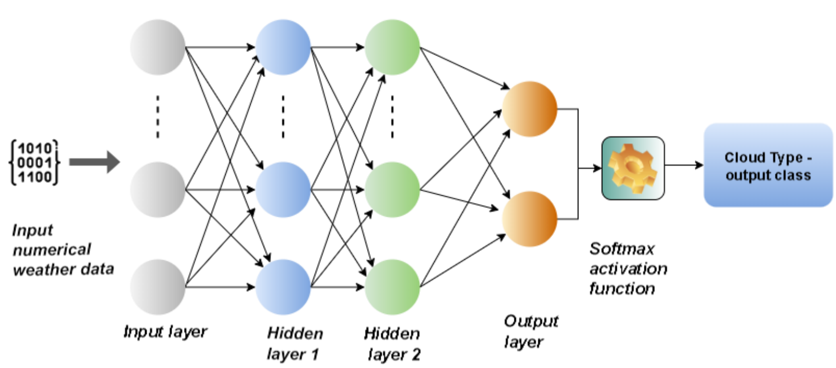

Fig. 4 MultiLayer Perceptron (MLP) model

not in the opposite direction. Numbers of hidden layers are generally depend on the dataset size and the objective of the problem. After the output layer as activation function softmax is use for multi level classification problems. Then it gives the output of the class type. In this project MLP was implemented to identify cloud types by using numeric data of the weather. The detailed model building crucial steps are explained in the implementation section.

4.2 Model 2 - CNN based image classification transfer learning methods

Convolutional neural networks (CNN) is the most established algorithm used in computer vision and image related research. Still, it is the state of the art image classification algorithm. CNN architecture consists of mainly three types of layers – convolutional layer, pooling layer and fully connected layer. These layers are used multiple times to build a CNN model. The fundamental layer is the convolutional layer which extracts the features of the input images. The output of the linear operation is then passed through a non-linear activation function. The pooling downsamples the dimension of the feature maps and decreases the number of learnable parameters. Max pooling and global average pooling are very common types. The output layer is then flattened to a 1-D array and connected to fully connected layers. Fully connected layers or dense layers can be one or more. It takes the output of a single vector and gives the probability distribution of each label or class. The last activation function gives the outcome of the class. In this way, a typical CNN-based image classification algorithm works.

CNN based transfer learning method uses pretrained algorithms on large image dataset to train and predict on a smaller dataset. The major advantage of using this kind of method is, that it can perform satisfactorily with a small image dataset with pre-existing knowledge and is very less resource-consuming to implement in real-time. In this project, the image dataset consists of 784 image files which are very fit for using transfer learning methods. As discussed earlier in the literature review section ‘VGG-16’ (Dey Roy et al., 2021) and ‘MobileNet’ (Tsukahara, Fudeyasu and Fujimoto, 2020) proved satisfactory results in the weather prediction. In this research, the VGG-16 was implemented separately and MobileNet was used in the multi-modal architecture.

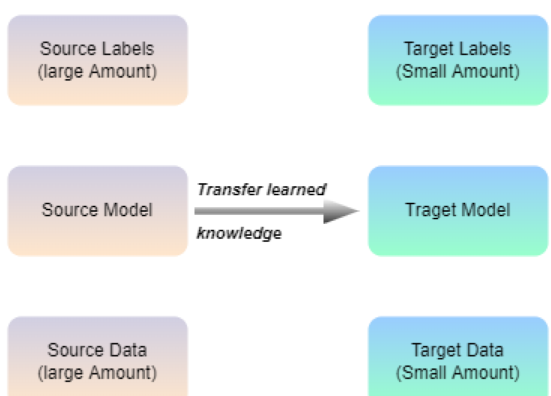

Fig. 5 Transfer learning method

4.3 Model 3 - Hybrid multi-modal deep learning model

A hybrid multi-modal model capable of handling multiple mixed input data. Here cloud image data and numeric weather data both can be used to build a model to classify cloud type. Implementation was carried out by Keras Functional API[4] methodology. Fig. 8 illustrates the model architecture implemented in this study. To efficiently extract the features of cloud images, CNN based pre-trained model named 'MobileNet' was used. To deal with numerical data MLP was used. The output from the last layer of these two models was merged in the 'concatenation' layer. In this layer, output tensors from MLP and MobileNet layers act as input. The tensors of the same shape from two layers were merged and returned a single tensor. Then the compiled model was connected with multiple dense layers with an activation function. In the end, the 'softmax' activation function was used to get the classification result. The final output had the class value of the cloud. This approach was unique as all the weather parameters were also taken into consideration along with image data to predict cloud patterns.

5. Implementation

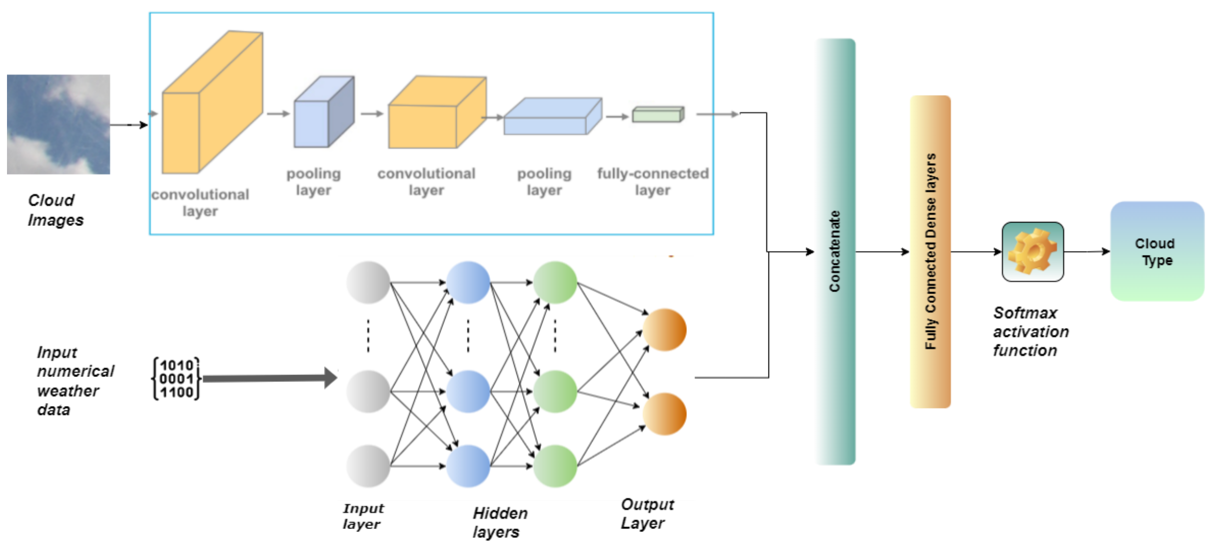

Fig. 6 Hybrid muti modal architecture

Crucial implementation steps during building the models are discussed in this section. Parameters and techniques related to specific algorithms are discussed briefly. Before implementing the algorithms, data preparation was done which is already discussed in the data preprocessing section. At first development environment was set up on a local PC and in google collab. Implementation was done in Python programming language with the help of different statistical, deep learning, and machine learning libraries.

[4] https://keras.io/guides/functional_api

5.1 API Integration to fetch weather data

Weather data preparation from image date and time was an important stage to start the project. Fig. 7 shows the data flow and how data preparation was carried out in different stages. The SWIMCAT dataset had CSV files for each type of cloud which were merged and date, time , exact coordination of Singapore were stored in variables. By those variables, the weather API was called. 'VisualCrossing' weather API was used here. Before using it was necessary to sign up in the API web portal to get the historical API endpoint URL. The output of the API response was parsed and stored in a CSV file. This whole implementation was done in the local environment with Jupiter Notebook.

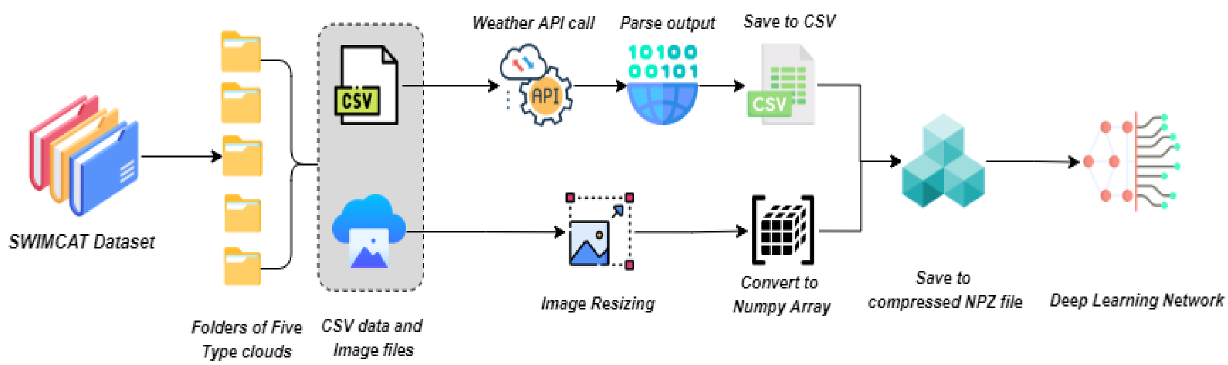

Fig. 7 Data preparation and transformations

5.2 Model 1 - Multi-layer perceptron (MLP)

MLP was implemented to predict cloud type by only using the numerical weather parameters which were fetched by API. The whole implementation was done in the Google Collab cloud environment.

- The dataset was uploaded in G-drive beforehand. The dataset had 8 weather features and one target column of cloud type with 784 rows. All the columns were normalized and then dataset was divided into train, test, and validation with 70%, 15%, and 15% respectively of total data.
- The target variable had text categorical data of cloud type. It was converted to multi-level numerical data of 5 types of integer values by using 'LabelEncoder' class of 'SciKit' package.
- MLP model was built by Keras sequential API. The input layer had 8 nodes or features, so the input shape was (8,). It must be a 1-d tensor.
- After the input layer there was a stack of hidden layers. Hidden layers were consisting of two dense layers and two dropout layers. Dropout layers were introduced carefully in the model to mitigate the risk of overfitting due to the small dataset. A first dense layer of 256 nodes was added followed by dropout (0.5). The second dense layer had 128 nodes followed by a dropout layer (0.5). Then a third dense layer of 64 nodes was added followed by a dropout layer (0.2). 'ReLu' non-linear activation function was used in all dense layers.

- In the output the dense layer had 5 nodes as data had 5 cloud class types. As it was a multi-level classification so 'softmax' activation function was used which gave the final value of cloud type.
- Model was compiled with 'categorical crossentropy' as loss function, 'adam' with learning rate 0.0001 as optimizer and accuracy as standard metrics.
- Early stopping technique was used during training to stop overfitting. The model was trained with 40 epochs and batch size of 64. In this way, the model was implemented step by step.

5.3 Model 2 - CNN based image classification transfer learning VGG-16

CNN based VGG-16 transfer learning model is implemented to classify the cloud images. As the image dataset was a small one so adopting a transfer learning method made sense.

- VGG-16 model was pre-trained on 'imagenet' dataset and used as a feature extractor in this project. The speciality of this model is the small convolutional filter size (3 × 3). It consists of 13 convolutional layers of 3×3 with stride size 1. The five max pooling layers had a stride size of 2. In the end, the model had two fully connected layers and an output dense layer.
- The dataset was uploaded in G-drive beforehand. Image size was converted to an input size of 226×226. The image data files from five folders were merged into one NumPy array dataset. Then all were transformed into compressed NPZs files for ease of computation. Then total data was divided into the train (70%), test (15%) and validation (15%) batch.
- Keras sequential API was used to implement the algorithm. Extra layers were added to the existing VGG-16 layers according to the understanding of the dataset. After 4 convolutional and max-pooling layers, flattened layer is used to convert the output to a single dimension vector. Two dense layers were added followed by the final output dense layer with 'softmax' activation function.
- Model was compiled with 'categorical crossentropy' as loss function, 'adam' with learning rate 0.0001 as optimizer and accuracy as standard metrics. Early stopping was used during training to avoid overfitting. The model was trained with 50 epochs and a batch size of 32. In this way, the model was implemented step by step.

5.4 Model 3 - Hybrid multi-modal deep learning model

This is the final model implemented to satisfy the objective to use both numeric data and image data for cloud classification. For this purpose, Keras functional API technique was applied.

- Data preparation is very vital to building this type of model. The first step was to merge all numeric data of clouds into one dataframe. Then NPZ files were created which contained both image and weather information. Then three data frames were created, one had all the numerical data, the second one had image NumPy arrays and the other one was the respective cloud type as the target variable. Then the dataset was shuffled and divided into a train (15%), test (15%) and validation (15%) datasets. The target variable had text categorical data of cloud type. It was converted to multi-level numerical data of 5 types of integer values by using 'LabelEncoder' class of 'SciKit' package.
- To handle numerical data in the network previously described MLP model is used without the classification head. Here the last dense layer had 10 nodes with 'relu' activation function.

- To extract the features of the images 'MobileNetV2' a pre-trained model was used. It is also a CNN based very lightweight algorithm. Depthwise Separable Convolution layers were used which reduced the complexity and model size.
- Two dense layers of 256 and 128 nodes were added with the 'Relu' activation function followed by one dropout layer of 0.2. Again two dense layers of 64 and 10 nodes were added.
- So the last layer of both the models had the same tensor shape. Both the output layers were merged in the concatenation layer. In this layer, actual magic happened with the help of the Keras Functional API method. This layer produced the output combining extracted features by MLP and MobileNetV2 models.
- After this a dense layer of 10 nodes with Relu non-linear activation function was applied followed by the final output layer of 5 nodes with a softmax activation function.
- Then after defining the model, it was compiled. Categorical cross entropy, adam, and accuracy were the loss function, optimizer and metrics respectively for compilation purposes. The model was fit with 50 epochs and a batch size of 10. An early stop mechanism with patience level 3 was used to avoid overfitting of validation data. This was a very GPU-heavy operation and carried out in google collab environment.

6. Evaluation

6.1 Model 1 - Multi-layer perceptron (MLP)

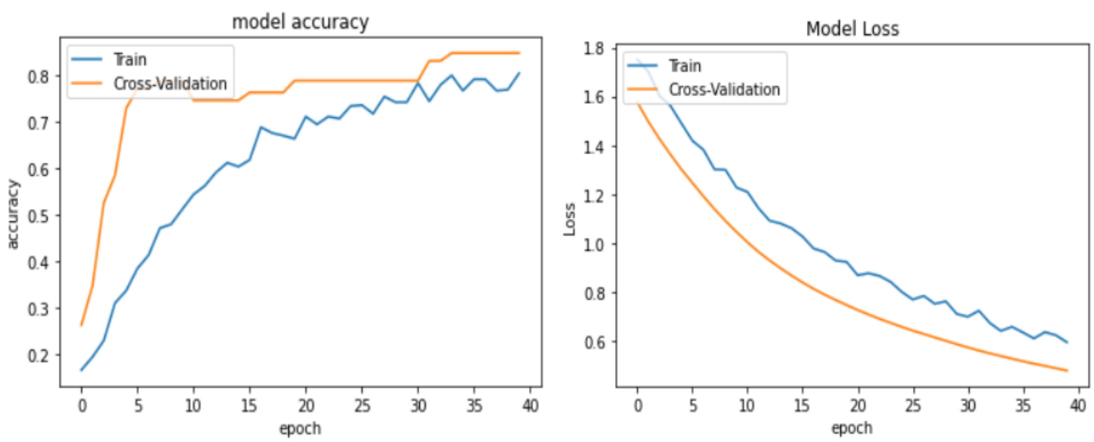

Fig. 8 Accuracy and loss curve of MLP model

Fig. 8 illustrates learning curve of accuracy and loss of MLP model. X-axis represents number of epochs. Y-axis represents accuracy and loss. Yellow line shows the validation and blue line shows training learning rate. The training accuracy of the model increased slowly first then after 20 epochs it showed good learning. On the other hand, validation learning rate improved drastically at first stage of 6 epochs and reach the best accuracy at 40 epochs. The Training accuracy and validation accuracy improved as epochs increased. That's why the loss was decreasing slowly. The hyperbolic curve of validation loss denotes that constant loss was observed during training. Around 40 epochs it stopped learning as there was no scope for learning. From Fig 9 the classification report shows overall 84% accuracy in cloud classification. It is important to understand whether cloud types were classified with what precision. The precision of the model is 0.85 which is a satisfactory score. The model was able to predict thick cloud type (Class type 2) most accurately whereas veil type cloud had the least precision.

	precision	recall	f1-score	support
0	0.81	1.00	0.90	30
1	0.88	1.00	0.94	36
2	1.00	0.69	0.82	13
3	0.80	0.42	0.55	19
4	0.76	0.80	0.78	20
accuracy			0.84	118
macro avg	0.85	0.78	0.80	118
weighted avg	0.84	0.84	0.82	118

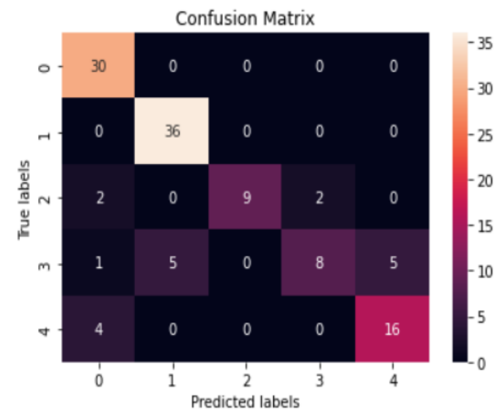

Fig. 9 Classification report and confusion matrix of MLP model

6.2 Model 2 - CNN based image classification transfer learning VGG-16

The fig. 10 shows the accuracy and loss curve for VGG-16 model. There was shape drop in the loss while training the data which indicates the faster learning rate of the developed model. Pretrained models have the advantage that they can learn new image patterns quickly. The validation loss decreased slowly and stabilized after around 20 epochs. The accuracy curve shows model stopped learning after 35 epochs and stopped there. Fig. 11 shows the average accuracy of the classification was 88%. From the confusion matrix, it can be observed that clear sky (class type 0) had the best classification accuracy and thick white sky (class type 3) had the highest misclassification rate. Overall VGG-16 perform better than the MLP model and outperformed by 4% classification accuracy.

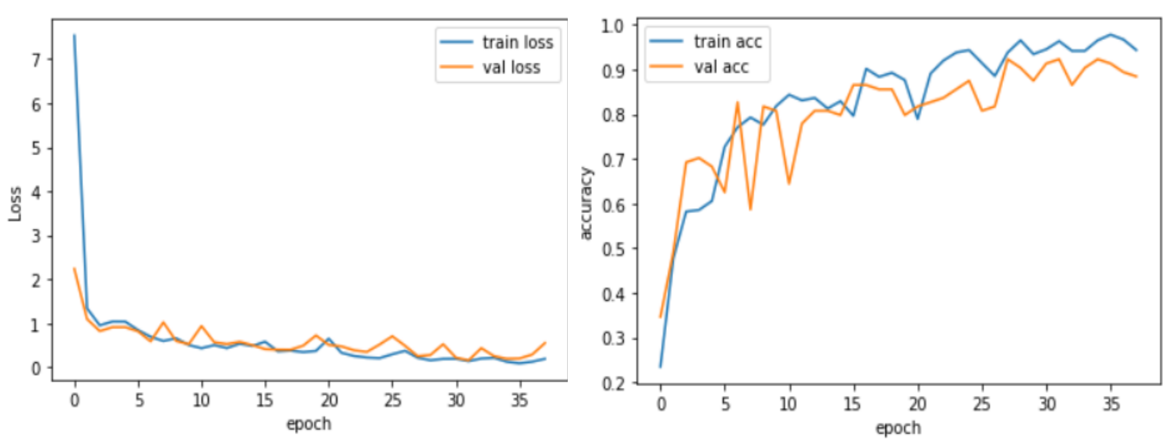

Fig. 10 Accuracy and loss curve of VGG16 model

	precision	recall	f1-score	support
0	0.96	1.00	0.98	27
1	0.83	0.97	0.90	36
2	1.00	0.71	0.83	14
3	0.69	0.75	0.72	12
4	1.00	0.73	0.85	15
accuracy			0.88	104
macro avg	0.90	0.83	0.86	104
weighted avg	0.90	0.88	0.88	104

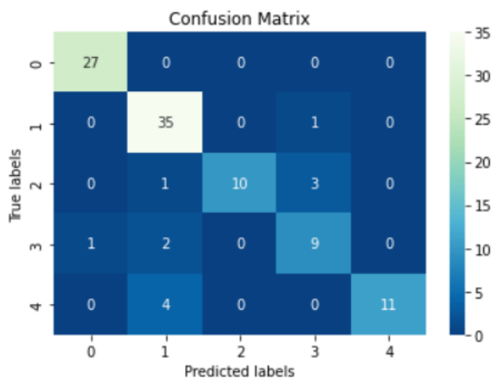

Fig. 11 Classification report and confusion matrix of VGG16 model

Model 3 - Hybrid multi-modal deep learning model

From the learning curve of Fig. 12 it can be observed that learning rate is very high, more than previously implemented models. At only 16 epochs the model was able to minimize the loss. After 6 epochs model stabilized in validation data and for training data the curve shows a smooth learning rate. The accuracy of training data increased as the epoch increased. The accuracy fluctuated initially in the case of validation data but after 8 epochs continuously increased and achieved the highest accuracy at the 16th epoch. The classification report suggested overall

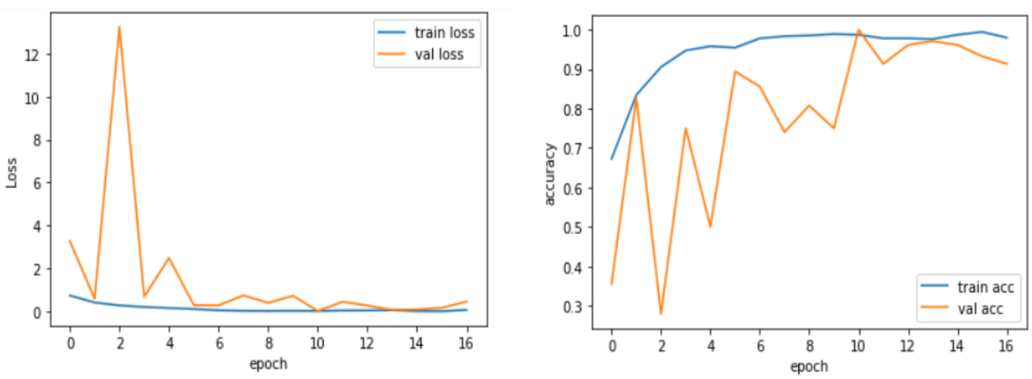

Fig. 12 Accuracy and loss curve of hybrid model

	precision	recall	f1-score	support
0	0.89	1.00	0.94	31
1	1.00	1.00	1.00	35
2	1.00	1.00	1.00	8
3	0.69	1.00	0.82	9
4	1.00	0.62	0.76	21
accuracy			0.92	104
macro avg	0.92	0.92	0.90	104
weighted avg	0.94	0.92	0.92	104

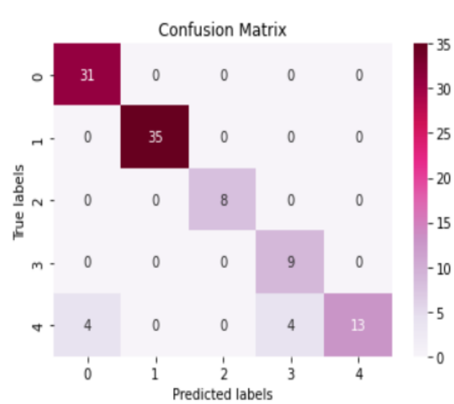

Fig. 13 Classification report and confusion matrix of hybrid model

classification accuracy of 92% which outperformed the previously built two models. The classification report of Model 3 (Hybrid) in figure 5 shows class 1 and class 2 had precision, recall, and f1-score of 1.00. This means the model trained on train data was able to identify all the images of class 1 (Patterned cloud) and 2 (Thick dark clouds) accurately on the test data. 35 images of patterned clouds and 8 images of thick dark clouds were taken for testing and no single image was misclassified. In the case of class 4, the precision was 1 which means there was no false positive so all the thick white clouds were predicted correctly as thick white clouds. Recall and f1-score for this class were 0.62 and 0.76 respectively. This indicates thick white clouds were wrongly predicted into different categories which generated false negatives. The confusion matrix shows out of 21 images of thick white clouds 13 were classified correctly but 4 were misclassified into class 3 (Veil clouds) and the rest 4 were misclassified as class 0 (Clear sky). This model's overall precision and recall were the same, 0.92. This hybrid model's confusion matrix showed 9 cloud images were misclassified among 104 images in the validation set. All the cloud types had good classification precision. So incorporating weather features along with images improved the model's performance. The hybrid model outperformed VGG16 by 4% accuracy and the MLP model by 8% accuracy which was commendable.

7. Discussion

The research was successfully implemented and achieved overall satisfactory results. The main objective was to classify cloud images into five categories. From different deep learning techniques, carefully suitable algorithms were chosen that already proved good output in the weather research domain. Dataset availability was a major concern for researchers in this domain, as there are very few public datasets available with labelled cloud images and with a time stamp. The dataset was small for implementing deep learning techniques, that's why dropout layers were added to the neural networks appropriately. Dey Roy et al. (2021) worked on the same dataset with CNN-based transfer learning algorithms that proved to be efficient in image classification. In this study, the pre-trained VGG16 model was implemented with added dense and dropout layers to classify cloud images. This model achieved 88% accuracy. The MLP model was the conventional way to predict weather using numerical data. It achieved 84% accuracy but was not able to predict dark cloud images efficiently and after trying several network architectures the algorithm overfits sometimes. So only using numerical weather variables from a location to go for cloud-type detection was not a good idea.

Table 2 Comparison of three models

Model Name	Precision	Recall	F1-score	Validation Accuracy
VGG-16	0.90	0.83	0.83	88%
MLP	0.85	0.78	0.80	84%
Hybrid	0.92	0.92	0.90	92%

But incorporating the weather variables in the image classification model did a great job. The multi-modal algorithm combining feature extraction by MobileNet and MLP achieved the highest accuracy of 92%. MobileNet performed well as a lightweight pretrained model. Table 2 illustrates the results of the three models implemented in this work. The hybrid architecture was influenced by the work of Tsukahara, Fudeyasu and Fujimoto (2020) on weather data and satellite images. In this study ground based on images and 9 weather parameters were used. The accuracy of the hybrid model was 4% greater than VGG16 and 8% better than the MLP model. So, weather parameters boost the feature extraction process while used with image data. The objectives of the study were quite satisfactory with good results. As the dataset was small so neural networks tended to overfit the data. To mitigate these dropout layers were used carefully, and an early stopping technique was employed at the time of training. The class imbalance was handled by using data augmentation techniques in image classification but still did not get the best results. There are scopes for improvements in the multi-modal architecture as well. Adding more numerical weather parameters can improve the prediction. A large dataset of cloud images can generate a better result using multi-modal architecture.

8. Conclusion and Future Work

Cloud pattern recognition is a popular domain of research in meteorology. Cloud formation and patterns depend on complex interactions of various parameters of the weather. In this study, an open source cloud image dataset was taken and weather parameters at the time of capturing the images were fetched by weather API to build a numerical dataset. Deep learning models were implemented in both datasets to classify cloud types. MLP model based on only numerical weather parameters did not perform well. Transfer learning-based VGG16 pre-trained model was used to classify cloud images which achieved a good accuracy of 88%. The multi-modal architecture incorporating weather features and cloud images did a really good job classifying images with 92% accuracy. The hybrid model was built with a pre-trained image classifier named MobileNet and MLP architecture to handle numerical weather features. It showed the power of feature extraction in classification using weather parameters accompanied by image data.

Minor problems faced during training of the model. The small dataset was one of the main reasons why class imbalance and overfitting issues were not able to handle efficiently. There are plenty of scopes to research in this domain. This study can be extended by adopting a large dataset of images of a bigger area like a whole district or province. More weather features can be identified and added to the multimodal architecture. Though the CNN based pre-trained image classification models generally produce satisfactory results still newly introduced transformer-based image classification algorithm, vision transformer can be a good area to explore in this domain.

9. Acknowledgement

I would like to thank my research guide Mr. Vladimir Milosavljevic for his continuous advice to improve my thesis. His dedication to his students is commendable. I also like to thank Mr. Keith Brittle for guiding us to write citations and reports correctly from the beginning. Thank you NCI library to create resources for thesis writing and finding relevant research papers. Thank you to my friends for sharing knowledge and family for continuous mental support during this period.

References

Ahrens, C. and Henson, R., 2016. Meteorology today. 12th ed. Boston, MA: Cengage Learning, pp.152-179.

Bahdanau, D., Cho, K. and Bengio, Y., 2014. Neural Machine Translation by Jointly Learning to Align and Translate. [online] arXiv.org. Available at: (https://arxiv.org/abs/1409.0473)

Buisán, S., Smith, C., Ross, A., Kochendorfer, J., Collado, J., Alastrué, J., Wolff, M., Roulet, Y., Earle, M., Laine, T., Rasmussen, R. and Nitu, R., 2020. The potential for uncertainty in Numerical Weather Prediction model verification when using solid precipitation observations. Atmospheric Science Letters, 21(7).

Charney, J.G., Neumann, J.V. and Fjoertoft. R.,1950. Numerical integration of the barotropic vorticity equation.Tellus 2, pp. 237–254

Coiffier, J. and Sutcliffe, C., 2011. Fundamentals of numerical weather prediction. Cambridge, UK: Cambridge University Press.

Dev, S., Lee, Y.H. and Winkler, S., 2015, September. Categorization of cloud image patches using an improved texton-based approach. In 2015 IEEE international conference on image processing (ICIP) (pp. 422-426). IEEE.

Dey Roy, S., Mohanta, A., Das, D. and Bhowmik, M., 2021. Cloud Pattern Classification for Rainfall Prediction using Convolutional Neural Network. 8th International Conference on Networking, Systems and Security,

Evans, B., Al-Sahaf, H., Xue, B. and Zhang, M., 2018. Evolutionary Deep Learning: A Genetic Programming Approach to Image Classification. 2018 IEEE Congress on Evolutionary Computation (CEC)

Gökçekuş, H., Kassem, Y. and Aljamal, J., 2020. Analysis of different combinations of meteorological parameters in predicting rainfall with an ANN approach: a case study in Morphou, Northern Cyprus. [online] 177, pp.350-362. Available at: https://www.deswater.com/DWT_articles/vol_177_papers/177_2020_350.pdf

Kurzrock, F., Cros, S., Ming, F.C., Otkin, J.A., Hutt, A., Linguet, L.,Lajoie, G. and Potthast, R., 2018. A review of the use of geosta-tionary satellite observations in regional-scale models for short-term cloud forecasting.Meteorologische Zeitschrift, 27, pp. 277–298

LeCun, Y., J., Yoshua, B., and Hinton, G., 2015. “Deep learning.” nature 521.753: 436.

Lynch, P., 2006. The emergence of numerical weather prediction. Cambridge: Cambridge University Press.

Mahajan, S. and Fataniya, B.,2019. Cloud detection methodologies: variants and development—a review. Complex & Intelligent Systems, pp. 251-261.

McCulloch W. and Pitts W., 1943. A logical calculus of the ideas immanent in nervous activity. Bull. Math. Biophys. 5, pp.115–133

Mittermaier, M., 2012. A critical assessment of surface cloud observations and their use for verifying cloud forecasts. QuarterlyJournal of the Royal Meteorological Society, 138, pp. 1794–1807

Moeeni, H. and Bonakdari, H., 2018. Impact of normalization and input on ARMAX-ANN model performance in suspended sediment load prediction. Water resources management, 32(3), pp.845-863.

Pallavi, V.P. and Vaithiyanathan, V., 2020. Combined artificial neural network and genetic algorithm for cloud classification. International Journal of Engineering Research & Technology , pp.787-794.

Poulsen, C., Egede, U., Robbins, D., Sandeford, B., Tazi, K. and Zhu, T., 2020. Evaluation and comparison of a machine learning cloud identification algorithm for the SLSTR in polar regions. Remote Sensing of Environment, pp248.

Rhee, E. and Phung, V., 2018. A Deep Learning Approach for Classification of Cloud Image Patches on Small Datasets.

Richter, M., Byttner, W., Krumnack, U., Wiedenroth, A., Schallner, L. and Shenk, J., 2021. (Input) Size Matters for CNN Classifiers. Lecture Notes in Computer Science, pp.133-144.

Rumelhart, D., Hinton, G. and Williams, R., 1986. Learning representations by back-propagating errors. Nature, 323(6088), pp. 533-536

S. Liu and M. Li, “Deep multimodal fusion for ground-based cloud classification in weather station networks,” Eurasip J. Wirel. Commun. Netw., vol. 2018, no. 1, 2018.

Šaur, D., 2015. Evaluation of the Accuracy of Numerical Weather Prediction Models. Advances in Intelligent Systems and Computing, pp.181-191.

Schizas, C.N., Michaelides, S.CH., Pattichis, C.S., and Livesay, R.R., 2002. Artificial Neural Networks in Forecasting Minimum Temperature. 112–114.

Sinko, M., Kamencay, P., Sykora, P., Benco, M. and Hudec, R., 2019. Cloud-type Classification of Ground-Based Images using Deep Learning. 2019 17th International Conference on Emerging eLearning Technologies and Applications (ICETA).

Strikwerda, J., 2004. Finite difference schemes and partial differential equations. Philadelphia, Pa.: Society for Industrial and Applied Mathematics (SIAM, 3600 Market Street, Floor 6, Philadelphia, PA 19104).

Tian, M., Chen, H. and Liu, G., 2019. Cloud Detection and Classification for S-NPP FSR CRIS Data Using Supervised Machine Learning. IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium.

Tsukahara, J., Fujimoto, Y. and Fudeyasu, H., 2020. Rainfall Forecasting by using Residual Network with Cloud Image and Humidity. 2020 IEEE 17th International Conference on Industrial Informatics (INDIN)

Zhang, J., Liu, P., Zhang, F. and Song, Q., 2018. CloudNet: Ground-Based Cloud Classification With Deep Convolutional Neural Network. Geophysical Research Letters, [online] 45.

Zhu, W., Chen, T., Hou, B., Bian, C., Yu, A., Chen, L., Tang, M. and Zhu, Y., 2022. Classification of Ground-Based Cloud Images by Improved Combined Convolutional Network. Applied Sciences.