Journal of Theoretical and Applied Information Technology
15th January 2023. Vol.101. No 1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

MULTI-CLASS WEATHER CLASSIFICATION USING MACHINE LEARNING TECHNIQUES

ADOLF FENYI [1], MICHAEL ASANTE[2]

Department of Computer Science and Information Technology, K.N.U.S.T, Ghana
Department of Computer Science and Information Technology, K.N.U.S.T, Ghana
E-mail: [1]adolffenyi@gmail.com, [2] masante.sci@knust.edu.gh

ABSTRACT

In the field of computer vision, multiclass weather classification from outdoor images is a difficult task to perform due to the diversity and lack of unique weather features. In this paper, a novel algorithm is formulated using machine learning techniques to classify several weather conditions such as sunny, cloudy, rainy, snowy and hazy. The researchers implemented a novel edge detection algorithm for the segmentation purpose while the Support Vector Machine was used for the classification task. However, before any classification is done, several weather features such as sky, cloud, rain streaks, snowflakes and dark channels are extracted from segmented images to increase the efficiency of the classifier. The extracted features are later concatenated and read into SVM for training and classification purposes.

The experiment revealed that multi-feature concatenation is essential since it yielded an average performance of 80.4% as compared to single feature selection of 72.8%.

From the evaluation of the proposed algorithm with other recent weather classification algorithms, the proposed algorithm exhibited an accuracy and a time complexity of (80.4%, $O(n^2)$) against the XGBoost (Extreme Gradient Boosting algorithm - 70.5%, $O(n^2)$), MLRA (Multilevel Recognition Algorithm - 75.1%, $O(n^3)$) and CNN (Convolution Neural Network - 83.9%, $O(n^4)$).

Keywords: *Edge detection; Weather classification; Support Vector Machine; Convolution Network Architecture; Object segmentation*

1. INTRODUCTION

Ever since the first digital computer was developed, there has been a significant improvement in weather predictions. Statistical models which were used in the past for weather predictions have become outdated to the extent that machine learning techniques have now dominated meteorological forecasting [1].

Multi-class weather classification is a fundamental and significant technique which has many potential applications, such as video surveillance and intelligent transportation. However, it is a challenging task due to the diversity of weather conditions.

Most of the methods used in weather classification are based on the assumption that the weather in an image is clear. However, different weather conditions such as rain, snow and haze could affect the quality of images. These effects could reduce the performance of a vision system which relies on image features. There are many applications of weather classification systems which detect and observe weather conditions and analyze features in images/videos [2]. A lot of images are available to the public from photo collections to video databases. These media require large amount of storage, computing power and memory, therefore, there is the need for efficient indexing and retrieval of such information. Classification plays an important and challenging role in this process.

In this research, the researcher aims at identifying the problems of classifying and representing multiple weather conditions such as sunny, rainy, snowy and hazy from a large number of colored images.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

1.1 Importance of Weather Classification

The following are some of the benefits derived from accurate weather predictions:
Transportation: All the different modes of transportations such as air, water and land are sensitive to weather changes. Marine engineers, marine biologists, ship captains, ecologist, fishermen and pilots depend highly on the weather. Therefore, an accurate weather prediction is necessary for aviation schedules, ocean navigations and fishing activities.
Agriculture: Farming activities are very important and depend heavily on the weather conditions. Rainfall, temperature and humidity are some of the factors that enable crops to grow very well. Accurate weather forecast can enable farmers to prepare their lands to overcome any unpleasant weather condition that can affect their crops.
Life: Accurate weather forecasting can save lives and properties through predictions of natural calamities such as heavy rainfall, cyclonic storms, hurricanes and tsunamis.
Social events: There are a lot of activities such as military operations, sports, geographical expeditions and social events that are organized only on accurate weather forecasting.

1.2 Problem Statement

Since the weather directly affect people's life, it is therefore necessary to accurately predict the weather conditions at a particular point in time. Currently, most of the weather prediction systems require series of sensors and manual assistance, but this cannot be provided in large quantities due to the high cost. This problem leads to inaccurate predictions.
Computer vision technology tends to remedy this problem by classifying the weather conditions through images. This solution leads to less cost and accurate predictions. However, most of the weather classification algorithms require high energy consumption due to the training and reasoning of the various classifiers. A typical example is the recent algorithm developed by Khan and Ismail using CNN with Keras Tensor framework [13]. Another example is the multilevel recognition system developed by Khan and friends [15].
Even though the recent algorithm developed by Padmini and Shankar [14] using XGBoost classifier require less energy, its accuracy is very low as compared to multilevel recognition system and CNN.
In this research, the researchers aim at classifying multiple weather conditions such as sunny, rainy, cloudy, snowy and hazy from a large number of colored images using less energy while maintaining high accuracy.

2. WEATHER CLASSIFICATION ALGORITHMS

In this section, the researchers will review some of the weather classification algorithms. These are summarized below:

2.1 Metric Learning Algorithm

In 2017, Fang Ju Lin and Tsai Pei Wang proposed a framework to investigate weather classification problems [3]. The framework is explained below:

Feature Extraction

Features such as sky saturation, hue and Discrete Cosine Transform were extracted from input images. These features are outlined below:

Sky feature

In most images, the sky region represents an essential area of the image. This region was identified using color-pair dictionary coding techniques. The researchers extracted adjacent pairs of pixels from the sky region. These pixels were learned using robust dictionary learning techniques [4] which resulted in a 6-D vector. The extracted sky features were found to consist of histogram values. These values were quantized separately for the various color components (L*, a*, and b*) where *L* represents *lightness*, *a* signifies *green-red* and *b* denotes *blue-yellow*. The first sky feature was quantized to generate 32 bins for each channel. These bins were later concatenated to generate a total of 96-dimensional features. In the second sky feature, the L component was quantized into 32 bins while the 'a' and 'b' were quantized into 16 bins each. These bins were later concatenated to generate 64-dimensional features.

Saturation feature

Saturation is defined by the transition from a pure color (100%) to gray (0%) at a constant lighting level. Since research has proven that the saturation level in cloudy images is lower than sunny images, the researchers used the saturation portion of the HSV color space to do their classification. The histogram generated from the saturated component was found to be very good.

Hue feature

Some features were also generated from the hue components of the images [5]. This was done by resizing an input image to 512 X 512. The resized image was later divided into sub-regions which highlighted all the essential features. The image was later converted to HSV color space where the hue component was extracted.

DCT coefficients feature

Discrete Cosine Transform converts an image from the spatial domain into frequency domain in order to generate efficient information. The 'L' component of every input image was resized into 50 X 50 before converting them into frequency domain using DCT.
After extracting the above features, it was concatenated into a long vector which was later normalized between 0 to 1.

Metric Learning

Since most of the distance-based classifiers such as the Euclidean Metric are suboptimal for classification [6], the ITML Metric learning algorithm was used which helped to overcome most of the problems associated with distance-based classifiers. This algorithm was introduced by Davis and friends [7]. Metric learning entails the act of generating a suitable metrics for a given set of data-points (pixels). Information Theoretic Metric Learning algorithm (ITML) characterizes the generated metrics using Mahalanobis distance function. It then learns the metric's parameters using Bregman's cyclic projection algorithm.

Experimental Result

The extracted features were evaluated using KNN (K-Nearest Neighbor), LMNN (Large Margin Nearest Neighbor) and the ITML (Information Theoretic Metric Learning) classifiers. Images used in the evaluation process were taken from the Sun Dataset, Labelme Dataset and Flickr. The minimum image dimension in the dataset was 600 while the maximum was 1500. Out of the 10,000 images used in the evaluation, 5000 were sunny while the remaining 5000 were cloudy. In order to make fair and precise comparison, the researchers performed five rounds of experiment. The results of the experiment are shown below:

Table 1: Normalized Accuracies Of Single Feature Sets Obtained By ITML

Features	**K = 11**	**K = 31**
Sky	64.8	66.9
Saturation	51.4	52.5
Hue	46.7	49.2
DCT	20.9	23.8

From the result in Table 1, it could be seen that the sky feature obtained the highest accuracy of 66.9%. The worst performance was recorded by the Discrete Cosine Transform at 20.9%. Table 1 also indicates that the higher the number of neighboring nodes selected, the greater the accuracy of the classification.

Table 2: Normalized Accuracies Of Single Feature Sets Obtained By The LMNN

Features	**K = 11**	**K = 51**
Sky	60.2	59.2
Saturation	50.0	51.8
Hue	46.1	46.4
DCT	18.4	23.5

Table 2 clearly outlines the classification accuracies obtained by the LMNN classifier. From the result, it could be seen that the performance of the sky feature was higher than the other features just as was seen in Table 1. The table also show that the number of neighboring nodes (*K*) selected has little effect on the classification accuracy. For example, the LMNN recorded an average classification difference of 1.53% between *K=11* and *K=51* while the ITML recorded an average difference of 2.15% between *K=11* and *k=51*.

Table 3: Normalized Accuracies Obtained From Feature Combination By SVM, KNN And ITML

Feature Combination	**Accuracy (%)**
Sky+Saturation+Hue+DCT+SVM	53.8
Sky+Saturation+Hue+DCT+KNN	63.5
Sky+Saturation+Hue+DCT+ITML	71.0

Journal of Theoretical and Applied Information Technology
15th January 2023. Vol.101. No 1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 3 outlines the accuracies obtained by the SVM, KNN and ITML when all the features extracted from the various images were combined. From the evaluation, a higher accuracy was generated using the ITML than the KNN and the SVM classifiers.

2.2 Convolution Neural Network Approach

In the paper published by Jehong An, Yunfan Chen and Hyunchul Shin, the authors combined Alexnet and Resnet with Multiclass Support Vector Machine [8]. Alexnet uses a total of 25 layers with Rectified Linear Activation function [9] while Resnet uses 347 layers for feature extractions. The two architectures are described below:

Alexnet

Alexnet [10] is made up of 5 convolutional and 3 fully connected layers. This architecture applies the softmax function for classification related problems. It is considered as the foundation of the Convolutional Neural Network architecture.

Resnet

Resnet [11] overcomes the problems of training deep neural networks through the introduction of residual blocks which is shown in Fig 1. In the figure, it is clear that there is a direct connection known as the 'skip connection' which skips some of the layers of the model. This connection changes the values of the output. For example, without the skip connection, the input (x) will be multiplied by the weight of the layers before a bias term and activation function are applied.

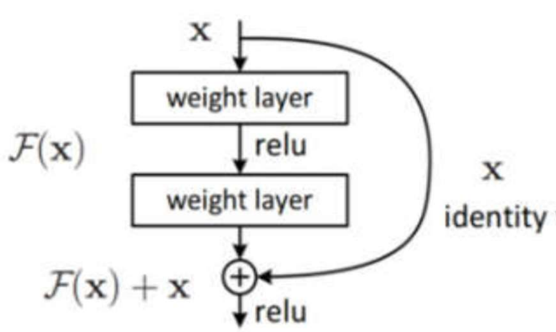

Figure 1: Resnet architecture

Multi-class SVM

The multi-class SVM was used to classify the images after Alexnet and Resnet had been used to extract features from these images. Support Vector Machine is part of the linear classifiers which include Logic Regression, Neural Network and Bayes classifiers.

Experimental Result

The dataset used for the classification were obtained from *weather database*, *desnownet* and *d-hazy datasets.*

In the *weather database*, 5000 images which consist of two classes (sunny and cloudy) were used in the evaluation. Out of the 5000 images, 70% were used for training while the remaining 30% were used for testing. The results of the evaluation are shown in Table 4.

Table 4: The Classification Accuracy Obtained By Alexnet And Resnet With Multiclass SVM

Image Samples	**Alexnet**	**Resnet**
Sunny	86	92
Cloudy	75	88

Table 4 outlines the classification accuracy obtained by both Alexnet and Resnet with multiclass SVM. From the table, it could be seen that a higher accuracy was obtained in Resnet than the Alexnet CNN architecture. The table also reveal that the performance of the sunny images was better than the cloudy images in both Alexnet and Resnet.
It was observed that image classification accuracy obtained from Resnet with Multiclass SVM was higher than Alexnet with Multiclass SVM classifier.

In the second evaluation, the performance of Resnet using *D-Hazy* and *Desnownet* datasets were evaluated. These two datasets contain two classes of images (snowy and hazy). Each class is made up of 1000 images of which 700 were used for training while the remaining 300 were used for testing purposes. The classification result is shown in Table 5.

Table 5: The Classification Accuracy Obtained By Resnet From D-Hazy And Desnownet Datasets

Images	Accuracy (%)
Snowy	95
Hazy	96

Table 5 outlines the performance of Resnet CNN architecture over D-Hazy and Desnownet datasets. From the evaluation, it could be seen that both the snowy and hazy datasets obtained almost the same performance, since the two datasets were separated by a difference of only 1%.

2.3 SAID Ensemble Method

In 2019, Oluwafemi and Zenghui proposed a novel framework called SAID (Selection Based on Accuracy Intuition and diversity) to classify outdoor weather images [12]. The researchers framework is described below:

Feature Extraction

The researchers extracted certain features from input images which are Hue, Saturation and Value (HSV), Gradient, Contrast and Local Binary Pattern (LBP). The extraction generated 128-dimension of LBP, 512- dimension of HSV, 128-dimension of gradient magnitude and 128-dimension of contrast features.

Model Selection

Classifiers selected based on accuracy, intuition and diversity for the ensemble learning. The minimum accuracy difference between the classifiers was set at 10%. The two researchers divided the experiment into two parts, so that each of the stacked ensembled algorithms can learn extracted features from input images. The first experiment consisted of KNN, RBF-SVM, and Random Forest algorithms while the second experiment was made up of Native Bayes, Random Forest and KNN. The accuracy for the model selection depended on cross validation with a fold setting of five. Features learnt from these stacked ensembled algorithms were merged together and classified using Gradient Boot meta classifier.

Experimental Result

This experiment was repeated ten times, and this produced the highest accuracy from the Random Forest algorithm at 84%. This was followed by RBF-SVM (70%) and Native Bayes (66%). The least accuracy was obtained from the K-Nearest Neighbor Algorithm at 58%. With respect to the ensembled learning algorithms, 'SAID Method I' obtained an accuracy of 85%, while 'SAID Method II' had 86%.

2.4 CNN with Keras-Tensor Framework

Sharma and Ismail recently developed a weather classification model using Convolution Neural Network with Keras Tensor framework [13] to classify the weather into four domains (cloud, rain, shine, and sunrise). Their model generated a validation accuracy of 94% against a validation loss of 22%.

2.5 XGBoost Classifier

Padmini and Shankar developed a weather classification model [14] to extract haze, fog and sunny images from the weather using a supervised classifier such as the XGBoost. Their method was implemented by generating a new dataset from several weather images collected from public databases. The new datasets were generated using psycho-visual analysis. From their evaluation result, the proposed algorithm generated an accuracy of 91.50% against the Naïve Bayes (77.90%), Decision Tree (84.99%), Support Vector Machine (89.52%) and AdaBoost (83%).

2.6 Multilevel Weather Recognition

Khan and friends developed a trajectory-level weather detection system [15]which has the capability of providing real time weather information using a single video camera. The researchers extracted two main features which include Histogram of Oriented Gradient (HOG) and Local Binary Pattern (LBP) from images. These extracted features were used as parameters to train several weather detection models using classifiers such as Gradient Boosting, Random Forest and Support Vector Machine. In the proposed model, the researcher merged three machine learning classifiers in a hierarchical structure instead of using a single classifier. The steps for the proposed model are outlined below:

- Train three separate detection models (Gradient Boosting, Random Forest and Support Vector Machine) for each level. Level 1 consists of clear, rain, snow, and fog features, Level 2 consists of light and heavy rain, Level 3 consists of light and heavy snow and Level 4 consists of distant

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

and near fog. In each of these levels, the best performing model is selected.

- Input the test image into the Level 1 model to obtain *temp* (contemporary weather feature). If the *temp* is clear, the final weather feature is clear; if *temp* is rainy, then pass the input image to Level 2 to get the final weather feature; If *temp* is snowy, then pass the input image to Level 3 to get the final weather feature and if *temp* is foggy, then pass the input image to Level 4 to get the final weather feature.

From the evaluation results, it was realized that the multilevel model provided an overall accuracy of 89.2% which is 3.2%, 7.5% and 7.9% higher than the performance obtained from Support Vector Machine, Random Forest and Gradient Boosting models respectively.

2.7 Edge Detection

Edge detection is applicable in image processing, computer vision and feature extraction. In an ideal case, applying an edge detector to an image leads to connected curves that signifies the image edges. Edge detection also reduces the amount of data a computer needs to process by eliminating unwanted information.

It is used in computer vision and image processing for feature detection. The goal of identifying variations in image intensities is to select important features or events in the object for decision making. Under ideal conditions, variations that result in image intensities are caused by discontinuities in image depth, surface orientation, material properties and scene illumination (SI).

Edge detection produces connected curves which represent the boundaries of an object. This indicates that it can reduce image size while maintaining its physical properties. Edges that are acquired from natural images suffer from fragmentation, missing segments and false edges [15]. These problems make data interpretation a difficult task [16].

There are several ways of performing edge detection which include Ant Colony Optimization [17], Statistical Range [18], Modified Moore Neighborhood [19], Wang and Li Operator [20], and Ghost Imaging [21].

Even though edge detection aids in image segmentation [22], data compression and image reconstruction, there are several problems connected with the existing edge detection techniques that weaken their capabilities. Some of these problems are the production of false and missing edges, errors in edge angle (EAG) estimation, poor localization and noise.

2.8 Support Vector Machine

Support Vector Machine (SVM) is a supervised machine learning algorithm which is used for classification. In SVM algorithm, each data is plotted as a point in n-dimensional space, where n signifies the number of features in the image [23]. Then, a classification is performed by computing the hyper-plane that separates the two classes [24]. The main goal of classification through a SVM is to identify an efficient way of learning good separating hyper planes in a hyper space.

The SVM is designed for binary classification. However, when dealing with multiple classes, the SVM can implements 'One against All' algorithm where *n* hyper planes are created. The *n* represents the number of classes in the dataset [25]. The SVM creates a number of support vector machine models based on the number of classes present in the dataset. It then treats each model as a binary classification system.

3.0 METHODOLOGY

The implementation of the proposed algorithm is divided into four main sections which are outlined below:

3.1 Segmentation

Segmentation is the process of dividing an input image into several parts which is known as image regions. The main aim of segmentation is to transform an image into something that is easier to understand and analyze. The segmentation algorithm used is described below:

- Convolve a set of 5 X 5 structuring elements with an input image.

$$\begin{bmatrix} 1 & 8 & 18 & 8 & 1 \\ 4 & 32 & 72 & 32 & 4 \\ 0 & 0 & 0 & 0 & 0 \\ -4 & -32 & -72 & -32 & -4 \\ -1 & -8 & -18 & -8 & -1 \end{bmatrix} \quad \begin{bmatrix} 1 & 4 & 0 & -4 & -1 \\ 8 & 32 & 0 & -32 & -8 \\ 18 & 72 & 0 & -72 & -8 \\ 8 & 32 & 0 & -32 & -8 \\ 1 & 4 & 0 & -4 & -1 \end{bmatrix}$$

Equation 1: Matrices for extracting horizontal and vertical pixels

$$\begin{bmatrix} 4 & 17 & 8 & 1 & 0 \\ 17 & 72 & 32 & 0 & -1 \\ 8 & 32 & 0 & -32 & -8 \\ 1 & 0 & -32 & -72 & -17 \\ 0 & -1 & -8 & -17 & -4 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 & 8 & 17 & 4 \\ -1 & 0 & 32 & 72 & 17 \\ -8 & -72 & 0 & 32 & 8 \\ -17 & -72 & -32 & 0 & 1 \\ -4 & -17 & -8 & -1 & 0 \end{bmatrix}$$

Equation 2: Matrices for extracting diagonal pixels at 45^o and 135^o

$$G(m,n) = \sum_{m=0}^{4}\sum_{n=0}^{4}(z(m,n) * t(m,n))$$

Equation 3: Convolution operation

where matrices *z* and *t* represent the input image and the kernel, * denotes the convolution operation

- Compute the magnitude of the gradient using Equation 4

$$h(m,n) = \sqrt{(Gx(m,n) + Gy(m,n) + Gz(m,n) + Gz1(m,n))^{2}}$$

Equation 4: Computation of the magnitude

Where Gx, Gy, Gz and Gz1 represent the gradients in the horizontal, vertical and diagonal directions at 45° and 135° respectively

- Suppress the impact of noise using the Gaussian equation

$$f(m,n) = \frac{1}{2\pi\sigma^2}e^{-\frac{m^2+n^2}{2\sigma^2}}$$

Equation 5: The Gaussian equation

- Partition the generated edges into a set of 5 X 5 matrices and estimate the minimum weighted variance in each local window for the segmentation process using Equation 7.

$$v = \frac{\sum_{i=0}^{N}(bg(i) - m)^{2} * fb(i)}{\sum fb(i)}$$

Equation 6: Computation of the variance

Where N represents the size of the pixels, m denotes the mean of the pixels, bg_i represents a pixel at index i and fb_i denotes the frequency of a pixel, bg_i

$$v_w = v * w$$

Equation 7: Computation of the weighted variance

Where v is the variance computed from Equation 7 and w is the weight of the pixels in each local window

- Produce Skeletal edges using Iterative parallel thinning algorithm [26].

3.2 Feature Extraction

For any successful vision recognition system, it is very important to select the right features to distinguish between images of the same scene. This issue involves analyzing several low-level features in the image which includes the sky, cloud, shadow, rain streaks, snowflakes, and dark channels. Proper analysis of these features will enable the classification algorithm to discover sunny, cloudy, rainy, snowy and hazy weather conditions in an image.

3.2.1 Sky feature

The sky feature is used to identify sunny weather in images. One of the characteristics of sunny images is that it has either a clear sky or a strong shadow while other images have a gray sky or a faint shadow. The researchers identified the sky region using the method proposed by Lu and friends [27]. In this method, the researchers extracted 131 dimensional features (mean HSV color and SIFT descriptor) from some images with size 15 X 15. These features were learned using Random Forest classifier. The sky region was segmented through the implementation of graph cut algorithm on those features.

3.2.2 Rain Streaks

The researchers identified rain streaks using Histogram of Gradient (HOG) based matching method proposed by Zhang and friends [28]. In this method, some HOG templates in different angles were constructed. The researchers used the guided image filters to decompose each image into a low and high frequency parts, so that the rain streaks could be seen in the high frequency parts. The HOG feature representing the rain streaks were extracted from the high frequency components of the image.

3.2.3 Snowflakes

Snow is light and soft and can even fly anywhere in the environment anytime there is a heavy wind. The researchers identified snowflakes as a form of noisy pixels in the image. A pixel is considered as a snowflake if its intensity value *(x, y)* is expressed as *m + u,* where *m* is the mean intensity value of the entire image and *u* is any value greater than 1.
All the snowflake pixels extracted were combined to form a long dimensional feature vector.

3.2.4 Dark channel

Haze is considered as an atmospheric phenomenon where dust, smoke and other particles in the air prevent the sky from being seen clearly. Dark channel prior is one of the methods used in identifying haze in an image, since they are located in the regions of an image where the intensities are very low. In extracting the haze features, the researchers decided to divide the input image into several patches. The median intensity values found in the dark areas of the image was computed and concatenated to form a long dimensional feature vector.

3.2.5 Cloud

The researchers identified the cloud features using Hybrid thresholding algorithm proposed by Qingyong Li and friends [29]. According to the researchers, cloudy images can be divided into unimodal and bimodal groups. While unimodal images are composed of a single element such as sky or cloud, bimodal images are composed of multiple elements (both sky and cloud). When the histograms of these two groups of images are examined critically, it would be seen that unimodal images have a single peak with a small variance while bimodal images consist of multiple peaks with large variance. In extracting the cloudy features from the images, the researchers computed a ratio from the blue and red channels of the image to improve its visual contrast. A standard deviation was then estimated from the ratio input image for the classification of the image into unimodal and bimodal. Images with standard deviation greater than 0.03 were classified as bimodal while those below were seen as unimodal. In extracting the cloud features from the unimodal images, a mean (u) and standard deviation (d) were estimated from the normalized blue-red ratio values. A threshold (T) was then calculated from the formular $T=u+3d$.
Pixels in unimodal images with values greater than T were identified as cloud elements, while those below were seen as sky elements. For bimodal images, the researchers used the Minimum Cross Entropy techniques [29] to extract the cloud elements.

3.2.6 Feature Concatenation

The researchers decided to concatenate all the four features (sky, rain streaks, snowflakes, and dark channels) into a long vector for each image. This feature vector does not have a fixed size but varies based on the number of features extracted from a given image. For example, $fv = \{x_1, x_2\}, \{x_1, x_2, x_3\}, \{x_1, x_2, x_3, x_4\}$ where x_1 corresponds to sky, x_2 represents rain streaks, x_3 denotes snowflakes and x_4 signifies dark channel features.

3.3 Training and Turning

After an image is segmented and appropriate features are extracted, they are assigned to the Support Vector Machine for training purpose. The Radial Basis Function kernel is used to convert the data set into high dimensional spaces so that the accuracy of the classifier would be improved.

3.4 Testing

After the training and turning process is complete, the SVM is now ready to be used for the weather classification task. An input image will undergo all the sequential steps described above before it is finally classified into any of the four weather conditions (sunny, rainy, snowy and hazy).

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

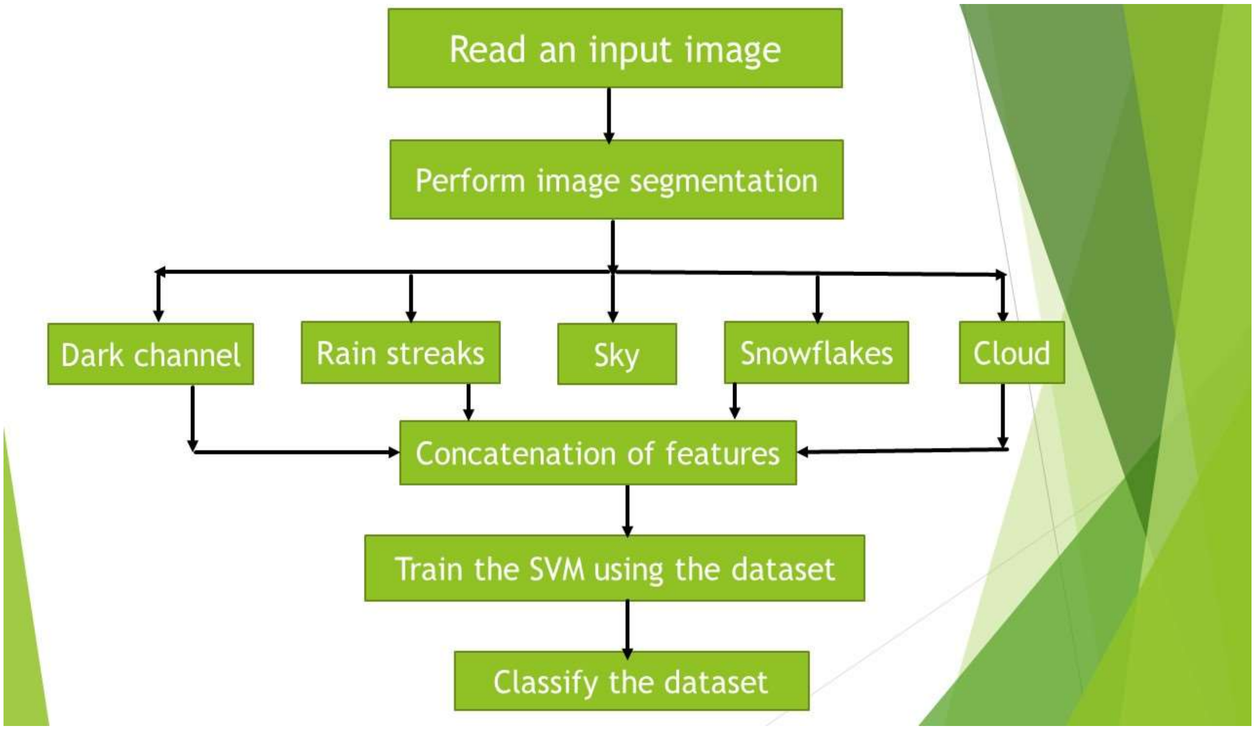

Figure 2: Flowchart Of The Proposed Algorithm

4.0 Results Analysis and Discussion

The above technique was implemented in MATLAB on an HP computer with 2.8 GHz CPU and 4GB of memory in which 5000 images of different weather conditions were used for the training and testing [30] under 495s. The various samples of the images are as follows: 1200 rainy, 1100 sunny, 1000 snowy, 950 cloudy and 750 hazy samples. 70% of each sample were used for training purpose while the remaining 30% were used for testing. Some of the sample images used are shown in Figures 3 and 4.

Figure 3: RGB Image Of Sunny Weather Condition

Journal of Theoretical and Applied Information Technology
15th January 2023. Vol.101. No 1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 4: RGB Image Of Rainy Weather Condition

The accuracy of the classifiers was estimated using Equation 8

$$Acc = \frac{X}{Y} * 100\ \%$$

Equation 8: Estimating the accuracy of the classifiers

Where X represents the correctly classified images and Y denotes the total test images

Tables 1 and 2 represent the performance of the proposed algorithm over single and multi-category feature sets while Table 3 shows the performance with other weather classification algorithms such as Convolution Neural Network [13], XGBoost classifier [14] and Multilevel Recognition algorithm [15].

To obtain the classification accuracies in Tables 5 and 6, the selected images were executed through a novel edge detection algorithm so that unwanted pixels (noise) could be eradicated from the images. Some low - level features such as the sky, cloud, shadow, rain streaks, snowflakes, and dark channels from the segmented images were also extracted afterwards. This process was essential since the SVM used the extracted features to properly classify sunny, cloudy, rainy, snowy and hazy weather conditions found in the various images.

During the classification process, the SVM was used to measure the accuracy of each individual feature extracted from the images using Equation 8. The results of the accuracies are found in Table 5. Since the researchers were not satisfied with the average accuracy of the various features, they decided to extract multiple features from a single image and later combined these individual features into a long vector. The classification accuracy is shown in Table 6. When the result in Table 6 is compared with Table 5, it would be realized that multi-feature concatenation provides better classification accuracy than single feature selection.

Table 5: Accuracies Of Single-Category Feature Set

Features	Accuracies (%)
Sky	79.6
Rain streak	72.1
Snowflake	68.1
Dark channel	69.9
Cloud	74.3
Average accuracies	**72.8**

Table 6: Accuracies Of Multi-Category Feature Sets

Features	Accuracies (%)
Sky + Rain streak + Cloud	77.5
Sky + Rain streak + Snowflake + Dark channel + Cloud	80.4

Tables 5 and 6 represent the accuracies obtained when the proposed algorithm was executed over the selected images. From the results in Table 1, it is clear that the proposed algorithm is weak in extracting only single features from images. This explains why an average accuracy of 72.8% was obtained. However, it exhibits amazing performance when multiple features are extracted and combined in an image. This performance is seen in Table 2. The Table also proof that the number of feature combinations is directly proportional to the accuracy of the algorithm, since three feature combination yielded an accuracy of 77.5% while all the five features yielded 80.4%.

Since multi-feature combinations yielded an impressive performance, the researchers will evaluate the output of the algorithm with other classification algorithms using this approach.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 7: Accuracies And Time Complexities Of The Various Classifiers

Classifiers	Sunny (%)	Cloudy (%)	Rainy (%)	Snowy (%)	Hazy (%)	Average Accuracy (%)	Time Complexity
XGBoost	77.8	69.5	67.5	63.6	74.2	70.5	$O(n^2)$
CNN	89.2	85.5	83.4	79.7	81.9	83.9	$O(n^4)$
MLRA	81.9	72.5	76.8	70.1	74.2	75.1	$O(n^3)$
Proposed algorithm	84.9	82.2	80.3	75.9	78.5	80.4	$O(n^2)$

Where MLRA represents Multilevel Recognition Algorithm [15], CNN denotes Convolution Neural Network Algorithm [13] and XGBoost signifies Extreme Gradient Boosting Algorithm [14].

In order to obtain the accuracies found in Table 7, all the test images were executed through the various algorithms that were used in the evaluation process. The algorithms were then used to determine the accuracies of the various image samples which has been recorded in Table 7.
The researchers went ahead to determine the time complexities of the various algorithms using the Big O notation from the programming codes.

From the result depicted in Table 7, it could be seen that the performance of the proposed algorithm was better than the XGBoost algorithm as a performance accuracy of 80.4% was obtained against 70.5%. This performance difference is due to the fact that the XGBoost algorithm used in the implementation of the weather recognition system was designed to classify only three weather features (sunny, haze and fog) in an image as compared to the proposed algorithm that deals with multiple weather features (rainy, cloudy, sunny, snowy and hazy). This also explains why the XGBoost algorithm obtained poor accuracy in rainy, snowy and cloudy weather conditions. From the evaluation of the running time, both the XGBoost and the proposed algorithm obtained the same time complexty of $O(n^2)$.

The proposed algorithm was also evaluated against the MLRA. From the evaluation, the MLRA obtained an accuracy of 75.1% against 80.4% from the proposed algorithm. The reason for this low performance is that the algorithm only merges three classification algorithms (Gradient Boosting, Random Forest and Support Vector Machine) without undertaking the input image through preprocessing stage (segmentation) where irrelevant information pertaining to the object of interest could be eradicated to ensure a higher accuracy. The concatenation of the three algorithms also explains why the time complexity of the algorithm was higher ($O(n^3)$) than the proposed algorithm $O(n^2)$. Even though the MLRA is recognized as multi-class weather recognition system, it is efficient on four weather conditions as compared to the proposed algorithm that handles five weather conditions.
The final evaluation was done against the Convolution Neural Network algorithm. From the evaluation results shown in Table 3, it was realized that the proposed algorithm had trailed behind the CNN since it obtained an accuracy of 80.4% against 83.9%. This performance difference of 3.5% is due to the number of layers implemented by the CNN architecture. The algorithm uses a total of 347 layers to extract several weather features from input images to ensure a better classification result. Even though this operation leads to a good accuracy, it also increases the time complexity of the algorithm. Table 3 clearly depicts that the time complexity of the proposed algorithm ($O(n^2)$) is better than the CNA ($O(n^4)$). Out of the four algorithms that were used in the evaluation, the researchers realized that there were only two algorithms (CNN and proposed) that could recognize all the weather conditions (sunny, cloudy, rainy, snowy and hazy) while maintaining a high accuracy.
Table 3 also reveals the accuracy of the individual weather features extracted by the various classifiers. From the Table, it could be seen that the sunny samples had the highest accuracy of 83.45% while the least was recorded in the snowy samples at 72.33%. For example, the proposed algorithm was able to obtain an accuracy of 84.9% for the sunny images while its recognition accuracy was 75.9% for the snowy samples. Similar performance was also recognized in the CNN algorithm.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

5. CONCLUSION

In this research, a novel algorithm is formulated to classify several weather conditions such as sunny, cloudy, rainy, snowy and hazy. The researchers implemented edge detection algorithm for the segmentation purpose while the Support Vector Machine was used for the classification. Before the SVM could classify any image, several weather features such as sky, rain streaks, snowflakes, clouds and dark channels were extracted to increase the efficiency of the SVM model.

In validating the implemented codes, the researcher utilized the Holdout Cross Validation System where 70% of the images were used for training while the remaining 30% were used for testing purposes. Even though this system is easy to implement, it is unsuitable for imbalanced dataset.

The researchers evaluated the performance of the proposed algorithm with RGB images having different weather conditions. From the experiment, the researchers realized that when multiple features are extracted and combined in a single image, the performance exceeds single feature selection. An average accuracy of 80.4% was obtained from multi-feature combination against 72.8% from single feature selection.

The researchers also evaluated the performance of the proposed algorithm with other recent weather classification algorithms. From the evaluation, the worst performance was recorded in the XGBoost classifier with an accuracy of 70.5%. Even though the Convolution Neural Network had an accuracy of 83.9% against the proposed algorithm of 80.4%, its running time was the poorest due to the numerous amounts of layers (347) implemented by the algorithm to extract essentials features from input images.

The researchers stated in the problem statement that high energy consumption is one of the challenges that affect computer vision weather recognition system. From the evaluation, it could be seen that the proposed algorithm is able to remedy this problem by utilizing less energy in the classification process. It is obvious that the performance of the proposed algorithm is promising.

6. FUTURE RESEARCH

Despite the low time complexity of the proposed algorithm, the research clearly shows that the algorithm has low classification performance when evaluated against the deep learning algorithms such as the Convolution Neural Network. In future, the researchers will improve upon the classification accuracy of the proposed algorithm while maintaining its time complexity. The researchers will also extend the proposed algorithm to recognize many weather features such as fog, frost, sleet and hail, since the current algorithm only recognizes five weather features (sunny, cloudy, rainy, snowy and hazy).

REFERENCES:

[1] N. Singh, S. Chaturvedi and S. Akhter, "Weather Forecasting Using Machine Learning Algorithm," in *International Conference on Signal Processing and Communication (ICSC)*, Noida - India, 2019.

[2] M. Elhoseiny, S. Huang and S. Elgammal, "Weather classification with deep convolutional neural networks," in *2015 IEEE International Conference on Image Processing (ICIP)*, Quebec City, QC, Canada , 2015.

[3] F. J. Lin and T. P. Wang, "Metric Learning for Image Weather Classification," *Multimedia Tools Application ,* pp. 12-45, 2017.

[4] S. Tariyal, A. Majumdar, R. Singh and M. Vatsa, "Deep Dictionary Learning," *Institute of Electrical and Electronics Engineers Access,* pp. 10096 - 10109, 2016.

[5] G. Hou, Z. Pan, B. Huang, G. Wang and X. Luan, "Hue Preserving-Based Approach for Underwater Colour Image Enhancement," *IET Image Processing,* pp. 5-12, 2018.

[6] K. Ganesan and H. Rajaguru, "Performance Analysis of KNN Classifier with Various Distance Metrics Method for MRI Images," *Soft Computing and Signal Processing,* pp. 673-682, 2019.

[7] J. Davis, B. Kulis, P. Jain and S. Dhillon, "Information-theoretic metric learning," in *Proceedings of the International Conference on Machine Learning*, California, USA, 2007.

[8] A. Jenong, C. Yunfan and S. Hyunchul, "Weather Classification Using Convolutional Neural Networks," in *International SoC Design Conference (ISOCC)*, Daegu, South Korea, 2018.

[9] A. Rasamoelina, F. Adjailia and P. Sinčák, "A Review of Activation Function for Artificial Neural Network," in *2020 IEEE 18th World Symposium on Applied Machine Intelligence and Informatics* , Herl'any - Slovakia, 2020.

[10] K. Alex, I. Sutskever and G. Hinton, "Imagenet classification with deep

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

convolutional neural networks," *Advances in Neural Information Processing Systems,* pp. 25-34, 2012.

[11] H. Kaiming, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," *Computer Vision and Pattern Recognition,* pp. 24-35, 2015.

[12] A. Oluwafemi and W. Zenghui, "Multi-Class Weather Classification from Still Image Using Said Ensemble Method," *Institute of Electrical and Electronics Engineers Xplore,* pp. 135-1239, 2019.

[13] A. Sharma and Z. S. Ismail, "Weather Classification Model Performance: Using CNN, Keras-TensorFlow," in *1st International Conference on Applied Computing & Smart Cities (ICACS21)* , Erbil - Iraq, 2022.

[14] T. N. Padmini and T. Shankar, "Static Weather Image Classification Based on Fog Aware Statistical Features Using XGBoost Classifier," *International Journal of Intelligent Systems and Applications in Engineering,* vol. 10, no. 3, pp. 64 - 72, 2022.

[15] S. Marikkannan and G. Kanagavalli, "Detection of Retinal Abnormality By Contrast Enhancement Method Using Curvelet Transform," *International Journal of Pure and Applied Mathematics,* pp. 1-8, 2017.

[16] T. Lindeberg, "Edge Detection and Ridge Detection with Automatic Scale Selection," *International Journal of Computer Vision,* vol. 30, p. 117–154, 1998.

[17] C. Martínez and M. Buemi, "Hybrid ACO Algorithm for Edge Detection," *Evolving Systems,* pp. 293-371, 2019.

[18] A. Bharodiya and A. Gonsai, "An improved edge detection algorithm for X-Ray images based on the statistical range," *Heliyon,* vol. 5, no. 10, pp. 1-9, 2019.

[19] S. Biswas and R. Hazra, "Robust Edge Detection Based on Modified Moore Neighborhood," *International Journal for Light and Electron Optics,* pp. 1-29, 2018.

[20] W. C. Lin and J. W. Wang, "Edge detection in medical images with quasi high-pass filter based on local statistics," *Biomedical Signal Processing and Control,* pp. 294-302, 2017.

[21] H. D. Ren, L. Wang and S. M. Zhao, "Efficient edge detection based on ghost imaging," *OSA Continuum 2,* pp. 47-56, 2019.

[22] Z. Zhang, H. Huazhu, D. Hang and S. Ling, "A Generic Edge-Attention Guidance Network for Medical Image Segmentation," in *International Conference on Medical Image Computing and Computer Assisted Intervention*, China, 2019.

[23] H. Wang, B. Zheng, S. Yoon and H. Ko, "A Support Vector Machine-Based Ensemble Algorithm for Breast Cancer Diagnosis," *European Journal of Operational Research,* pp. 689-695, 2018.

[24] G. Sun, T. Chen and C. Li, "Internet Traffic Classification Based on Incremental Support Vector Machines," *Mobile Networks and Applications ,* pp. 791-794, 2018.

[25] A. Chriki, H. Touati and H. Snoussi, "SVM-based indoor localization in Wireless Sensor Networks," in *13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, Valencia, Spain , 2017.

[26] B. Bataineh, "An Iterative Thinning Algorithm for Binary Images Based on Sequential and Parallel Approaches," *Pattern Recognition and Image Analysis,* pp. 34-43, 2018.

[27] C. Lu, D. Lin, J. Jia and C. K. Tang, "Two-Class Weather Classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 39, no. 12, pp. 2510 - 2524, 2017.

[28] Z. Zhang, H. Ma, H. Fu and C. Zhang, "Scene-free multi-class weather classification on single images," *Neurocomputing,* vol. 207, no. 8, pp. 365-373, 2016.

[29] Q. Li, W. Lu and J. Yang, "A Hybrid Thresholding Algorithm for Cloud Detection on Ground-Based Color Images," *Journal of Atmospheric and Oceanic Technology,* vol. 28, no. 2, pp. 1286-1296, 2011.

[30] A. Gbeminiyi, "Multi-class Weather Dataset for Image Classification," Mendeley Data, London, UK, 2018.