Journal of Engineering Sciences
Vol 12, Issue 05, MAY /2021
ISSN NO:0377-9254

Weather Prediction Using Deep Learning Techniques

[1]N.Priyanka,[2] Luis Sreeja, [3]Palli Manogna, [4]Uppu Niharika

N.Priyanka Assistant Professor in Department of IT Teegala Krishna Reddy Engineering College,Hyderabad,Telangana.

Luis Sreeja UG Scholar in in Department of IT Teegala Krishna Reddy Engineering College,Hyderabad,Telangana.

Palli Manogna UG Scholar in in Department of IT Teegala Krishna Reddy Engineering College,Hyderabad,Telangana,

Uppu Niharika UG Scholar in in Department of IT Teegala Krishna Reddy Engineering College,Hyderabad,Telangana

Abstract:

Extracting information related to weather and visual conditions at a given time and space is indispensable for scene awareness, which strongly impacts our behaviours, from simply walking in a city to riding a bike, driving a car, or autonomous driveassistance. Despite the significance of this subject, it is still not been fully addressed by the machine intelligence relying on deep learning and computer vision to detect the multi-labels of weather and visual conditions with a unified method that can be easily used for practice. What has been achieved to-date is rather sectorial models that address limited number of labels that do not cover the wide spectrum of weather and visual conditions. Nonetheless, weather and visual conditions are often addressed individually. In this paper, we introduce a novel framework to automatically extract this information from street-level images relying on deep learning and computer vision using a unified method without any pre-defined constraints in the processed images. A pipeline of four deep Convolutional Neural Network (CNN) models, so-called the WeatherNet, is trained, relying on residual learning using ResNet50 architecture, to extract various weather and visual conditions such as Dawn/dusk, day and night for time detection, and glare for lighting conditions, and clear, rainy, snowy, and foggy for weather conditions. The WeatherNet shows strong performance in extracting this information from user-defined images or video streams that can be used not limited to: autonomous vehicles and drive-assistance systems, tracking behaviours, safety-related research, or even for better understanding cities through images for policy-makers.

I. Introduction

Cities are complex entities by nature due to the multiple, interconnected components of their systems. Features of the physical environment extracted from images, or so-called urban scenes, have great potential for analysing and modelling cities because they can contain information on a range of

factors such as people and transport modes, geometric structure, land use, urban components, illumination, and weather conditions [1]. In recent years, computer vision techniques have shown progress in extracting and quantifying these features [2,3]. This article is concerned with the recognition of weather and visual conditions, which are two related but separate aspects of urban scenes that can be extracted in order to better understand the dynamics of the appearance of the physical environment [4]. In this study, we refer to visual conditions as the significant changes in the appearance of cities during dawn/dusk, day or night-time including the effect of glare on visibility, whereas weather conditions are the meteorological changes of the environment due to precipitation including clear, rainy, foggy, or snowy weather. They represent crucial factors for many urban studies including transport, behaviour, and safety-related research [5]. For example, walking, cycling, or driving in rainy weather is associated with a higher risk of experiencing an incident than in clear weather [5,6]. Fog, snow, and glare have also been found to increase risk [6,7]. Importantly, it is not only the inherent risk that different weather and visual conditions pose to human life that is of interest to researchers. Scene awareness for autonomous navigation in cities is highly influenced by the dynamics of weather and visual conditions and it is imperative for any vision system to cope with them simultaneously [8]. For example, object detection algorithms must perform well in fog and glare as well as in clear conditions, in order to be reliable. Accordingly, finding an automatic approach to extract this information from images or video streams is in high demand for computer scientists, planners, and policy-makers. While there are different methods that are used to understand the dynamics of weather and visual conditions, a knowledge gap appears when addressing this subject. To date, these two crucial domains—weather and visual conditions—have been studied individually, ignoring the importance of understanding the dynamics and impact of one domain on the other. There is no unified method that can extract information related to both weather and visual conditions from a street-level image that can be utilised by planners and policy-makers. Building on the current advances of scene awareness based on computer vision, in this paper, we present a novel framework, WeatherNet, that aims to recognise and map the dynamics of weather and visual conditions with a unified method.

The framework takes single-images as input and does not require pre-defined constraints such as the camera angle, area of interest, etc. WeatherNet relies on multiple deep convolutional neural network (CNN) models that aim to recognise visibility related conditions such as dusk/dawn, day or night-time, glare, and weather conditions such as clear, fog, cloud,

rain, and snow. The motivation behind WeatherNet is to practically extract and map weather information in cities that could help planners and policy-makers to analyse cities and contribute to the intelligent systems of navigation in cities and autonomous driving. Figure 1 shows the output of the WeatherNet framework.

II. PROBLEM STATEMENT

Based on current literature, there is still on-going research to cover the current limitation in addressing the weather and visual conditions simultaneously, in which addressing one domain only would not necessarily cover the dynamics of the appearance of urban scenes. For instance, cities may appear darker when it rains in the day-time than during clear weather at the same time. While the above-mentioned models show progress in the given tasks, there are number of knowledge gaps that needs to be addressed to cover the stated subject of weather and visual classification, which are:

1) These crucial domains- weather and visual conditions- have been studied individually, ignoring the importance of understanding the dynamics and impact of one domain on the other. There is no unified method that can extract information related to both weather and visual conditions from a street-level image;

2) The framework of the WeatherNet 4 weather classification has been treated with a limited number of labels, ignoring the variation of weather conditions. Even when weather is treated as a multi-label classification, a knowledge gap appears in representing scenes with multiple labels that simultaneously co-exist;

3) Current models used to classify weather and visual conditions are either limited to a presenting requirement or limited in accuracy. The methods are not up-to-date with the state-of-the-art of machine vision research (i.e. no models rely on residual learning to understand weather).

III. Proposed methodology

o address the current knowledge gap, we introduce a framework of parallel deep CNN models to recognise weather and visual

conditions from street-level images of urban scenes, so-called WeatherNet (See Figure 2). This WeatherNet comprises four deep CNN models to detect dawn/dusk, day, night-time, glare, rain, snow, and fog respectively. These models are:

1) NightNet detects the differences between dawn/dusk, day and night-time. It aims to understand the subtleties of street-level images despite the dynamics of weather conditions and urban structure,

2) GlareNet detects images with glare regardless of its source (sun or artificial light) for both dawn/dusk, day and night-time of various weather conditions. Glare is defined as a direct light source that can be seen to cause rings or star effect on the length of the camera without any correction,

3) PreciptationNet detects clear, rainy, or snowy weather for both day and night-time,

4) FogNet detects the occurrence of fog for dawn/dusk, day and night-time and whether this fog happens in the existence of clear, snowy, or rainy weather.

IV. Implementation

4.1 Gathering Data:

Data Gathering is the first step of the machine learning life cycle. The goal of this step is to identify and obtain all data-related problems. In this step, we need to identify the different data sources, as data can be collected from kaggle . It is one of the most important steps of the life cycle. The quantity and quality of the collected data will determine the efficiency of the output. The more will be the data, the more accurate will be the prediction.

This step includes the below tasks:

- Identify various data sources
- Collect data
- Integrate the data obtained from different sources

By performing the above task, we get a coherent set of data, also called as a dataset. It will be used in further steps.

4. 2. Data preparation

After collecting the data, we need to prepare it for further steps. Data preparation is a step where we put our data into a suitable place and prepare it to use in our machine learning training.

In this step, first, we put all data together, and then randomize the ordering of data.

This step can be further divided into two processes:

- Data exploration:

It is used to understand the nature of data that we have to work with. We need to understand the characteristics, format, and quality of data.

A better understanding of data leads to an effective outcome. In this, we find Correlations, general trends, and outliers.

- Data pre-processing:

Now the next step is preprocessing of data for its analysis.

4.3. Data Wrangling

Data wrangling is the process of cleaning and converting raw data into a useable format. It is the process of cleaning the data, selecting the variable to use, and transforming the data in a proper format to make it more suitable for analysis in the next step. It is one of the most important steps of the complete process. Cleaning of data is required to address the quality issues.

It is not necessary that data we have collected is always of our use as some of the data may not be useful. In real-world applications, collected data may have various issues, including:

- Missing Values
- Duplicate data
- Invalid data
- Noise

So, we use various filtering techniques to clean the data.

It is mandatory to detect and remove the above issues because it can negatively affect the quality of the outcome.

4.4 Data Analysis

Now the cleaned and prepared data is passed on to the analysis step. This step involves:

- Selection of analytical techniques
- Building models
- Review the result

The aim of this step is to build a machine learning model to analyze the data using various analytical techniques and review the outcome. It starts with the determination of the type of the problems, where we select the machine learning techniques such as Classification, Regression, Cluster analysis; Association, etc. then build the model using prepared data, and evaluate the model.

Hence, in this step, we take the data and use machine learning algorithms to build the model.

4. 5. Train Model

Now the next step is to train the model, in this step we train our model to improve its performance for better outcome of the problem.

We use datasets to train the model using various machine learning algorithms. Training a model is required so that it can understand the various patterns, rules, and, features.

4.6. Test Model

Once our machine learning model has been trained on a given dataset, then we test the model. In this step, we check for the accuracy of our model by providing a test dataset to it.

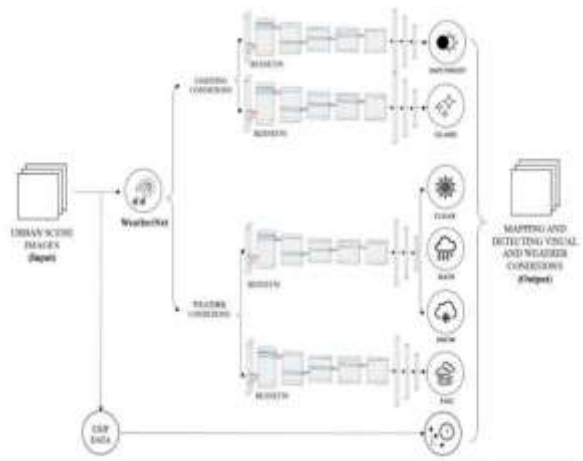
Testing the model determines the percentage accuracy of the model as per the requirement of project or problem.

4.7. Deployment

The last step of machine learning life cycle is deployment, where we deploy the model in the real-world system.

If the above-prepared model is producing an accurate result as per our requirement with acceptable speed, then we deploy the model in the real system. But before deploying the project, we will check whether it is improving its performance using available data or not. The deployment phase is similar to making the final report for a project.

V. Architecture Used



VI. Methodology

Methodology is the hypothetical, orderly research of the strategies applied to a zone of study. This incorporates the hypothetical investigation of the assortment of techniques and standards related with a surge of information. There are three kinds of strategies. They are: Explorative approach, Descriptive technique, Experimental philosophy. In our venture, the test approach is utilized.

6.1 Ensemble methodology:

It is a procedure of running various forecast models on a specific dataset and

consolidates into a solitary last expectation model. The primary thought is that, increasingly viable results are obtained when poor models are effectively consolidated. Most ordinarily utilized techniques are: Bagging, Boosting, Random woodlands, stacking.

6.2 Bagging:

Packing produces the best type of forecast by considering irregular information tests with substitution and of a similar size as that of the underlying dataset and regularly thinks about homogeneous students. At the point when a subset of the first information are taken, all the attributes are considered to parcel a hub. For each bootstrap information a model will be built for expectation, normal the models and experiences casting a ballot the most rehashed outcome will be the best one among all by decreasing forecast change and builds security.

6.3 Boosting:

In producing numerous single models and totaling their presentation, boosting is like sacking and arbitrary woodland, yet changes from them in refreshing the loads of results at every emphasis while over and over utilizing same unique dataset. The point is to iteratively fit models so that model preparing at a given stage relies upon the models fitted in the past stages. "Boosting" is the most well-known of these techniques, as it creates an outfit strategy which is generally less one-sided than the powerless students that make it up.

6.4 Random forest:

It is practically like packing, however the principle contrast is that in random forest, just a subset of highlights are picked indiscriminately from the aggregate and the best parting highlight from the subset is utilized to break every hub into a tree, though in sacking all highlights are required to break a hub. They likewise vary in the expansion of irregular examining of factors during the bootstrap information age process. Random forest can diminish the expectation fluctuation in sacking by this arbitrary testing. Random forest can decrease the expectation change in packing by this arbitrary examining. The model can be developed by random forest for every one of those bootstrap datasets and afterward at last produces a last model.

VII. Result Analysis

Ensemble approaches use different learning algorithms to achieve better predictive efficiency than either of the respective

learning algorithms alone could probably achieve. The ensemble and predictive algorithms like Random forest, Linear discriminate analysis, generalized boosted algorithm, Support vector machine are utilized.

Linear Discriminate Analysis or LDA is a dimensionality decrease procedure. It is utilized as one of the pre-processing step in Machine Learning and utilizations of pattern alignment. The objective of LDA is to extend the highlights in high to a low-dimensional space so as to stay away from the scourge of dimensionality and furthermore lessen assets and dimensional expenses. . LDA centers basically around anticipating the highlights in higher measurement space to bring down measurements. it can be accomplished in three stages: Firstly, you have to ascertain the distinctness between classes which is the separation between the mean of various classes. This is known as the between-class change. Furthermore, ascertain the separation between the mean and test of each class. It is likewise called the within-class variance. At long last, build the lower-dimensional space which boosts between class change moreover limits the inside class fluctuation. P is considered as lower-dimensional space projection, which is otherwise called as Fisher's model.

Generalized Boosting Models fit various decision trees to enhance the exactness of the template. For each new tree in the model, an arbitrary subset of the considerable number of information is chosen utilizing the boosting. For each new tree in the model the information is weighted so the data which is worthless demonstrated by previous trees consists of higher probability of being chosen in the latest prototype. This implication follows the principal tree is suited, the model will contemplate the misunderstanding with the assumption of that tree to apt the following tree, etc. By taking into consideration of the attack of past trees that are fabricated, the framework ceaselessly strives to improve its veracity.

SVM is a managed AI computation which can be employed for either classification or regression challenges. In the SVM calculation, every datum is plotted as a point in n-dimensional space with the estimation of each element being the expectation of a particular organizes. By then, the grouping is performed by detecting the hyper-plane that separates the two classes very well. Hyper-planes are decision confines that help

portray the data centers. Data focuses falling around either side of the plane can be credited to different classes. Similarly, the part of the hyper-surface depends on the amount of features. In case the amount of data features is 2, by then the hyper-plane is just a line. If the amount of data features is 3, by then the hyper-surface moves toward a two-dimensional plane.

VIII. Conclusion

In this paper, we presented a novel framework, WeatherNet, to detect and map weather and visual conditions from single-images relying on deep learning and computer vision. WeatherNetis capable of detecting 10 classes: dawn/dusk, day, night, glare, no glare, fog, no fog, clear, rainy, and snowy weather. We aimed to exemplify the application of deep learning and computer vision for scene-awareness and understanding the dynamics of the appearance of urban scenes that could be useful for autonomous applications in cities or elsewhere. After training four deep CNN models on street-level images from different corners of the globe of various urban structure, weather conditions, and visual appearances, the proposed WeatherNet showed a strong performance in recognising the combination of different categories of a single image. For example, by using the WeatherNet framework, urban scenes of street-level images can be classified with multiple classes for a given space and time such as 'image at daytime, with fog and no rain, in which glare exists'. The novelty of the proposed framework is in its simplicity for practical applications and for tackling various conditions in a binary fashion, relying on a unified method without pre-defined constraints for processing images. The proposed framework can be utilised for various proposes; it may be helpful for data automation and autonomous driving in cities, also, it may be utilised toward data automation for mapping and urban planning purposes. For future work, there are two main areas that seem promising to optimise and further validate the presented framework. First, experimenting with different architectures of CNN models including an attention-aware layer may enhance the overall performance of the model and allow further multi-task classification. Second, deploying the weights of the WeatherNet framework on a spatio-temporal image dataset that is fused with historical meteorological data could be used for further evaluation of the framework performance in a more practical setting.

IX. References

1. Narasimhan, S.G.; Wang, C.; Nayar, S.K. All the Images of an Outdoor Scene. In Computer Vision—ECCV 2002; Heyden, A., Sparr, G., Nielsen, M., Johansen, P., Eds.; Springer: Berlin/Heidelberg, Germany, 2002; Volume 2352, pp. 148–162, ISBN 978-3-540-43746-8.

2. Ibrahim, M.R.; Haworth, J.; Cheng, T. Understanding cities with machine eyes: A review of deep computer vision in urban analytics. Cities 2020, 96, 102481.

3. Ibrahim, M.R.; Haworth, J.; Cheng, T. URBAN-i: From urban scenes to mapping slums, transport modes, and pedestrians in cities using deep learning and computer vision. Environ. Plan. B Urban Anal City Sci. 2019, 239980831984651. [CrossRef]

4. Liu, Y.; Kang, J.; Zhang, Y.; Wang, D.; Mao, L. Visual comfort is affected by urban colorscape tones in hazy weather. Front. Archit. Res. 2016, 5, 453–465. [CrossRef]

5. Lacherez, P.; Wood, J.M.; Marszalek, R.P.; King, M.J. Visibility-related characteristics of crashes involving bicyclists and motor vehicles—Responses from an online questionnaire study. Transp. Res. Part F Traffic Psychol. Behav. 2013, 20, 52–58. [CrossRef]

6. Branion-Calles, M.; Nelson, T.; Winters, M. Comparing Crowdsourced Near-Miss and Collision Cycling Data and Official Bike Safety Reporting. Transp. Res. Rec. J. Transp. Res. Board 2017, 2662, 1–11. [CrossRef]

7. Dozza, M.; Werneke, J. Introducing naturalistic cycling data: What factors influence bicyclists' safety in the real world? Transp. Res. Part F Traffic Psychol. Behav. 2014, 24, 83–91. [CrossRef]

8. Narasimhan, S.G. Models and Algorithms for Vision through the Atmosphere. 2003. Available online: https://dl.acm.org/citation.cfm?id=997554 (accessed on 28 November 2019).

9. Lagorio, A.; Grosso, E.; Tistarelli, M. Automatic Detection of Adverse Weather Conditions in Traffic Scenes. In Proceedings of the 2008 IEEE Fifth International Conference on Advanced Video and Signal Based Surveillance, Santa Fe, NM, USA, 1–3 September 2008; pp. 273–279. 10. Middleton, W. Vision through the Atmosphere; University of Toronto Press: Toronto, ON, Canada, 1952.

11. Hautiére, N.; Tarel, J.-P.; Lavenant, J.; Aubert, D. Automatic fog detection and estimation of visibility distance through use of an onboard camera. Mach. Vis. Appl. 2006, 17, 8–20. [CrossRef]

12. Negru, M.; Nedevschi, S. Image based fog detection and visibility estimation for

driving assistance systems. In Proceedings of the 2013 IEEE 9th International Conference on Intelligent Computer Communication and Processing (ICCP), Cluj-Napoca, Romania, 5–7 September 2013; pp. 163–168.