The Netherlands Press

Journal of Airline Operations and Aviation Management

Article

HIGH DIMENSIONAL WEATHER DATA USED IN A DEEP GENERATIVE MODEL TO PREDICT TRAJECTORIES OF AIRCRAFT

Umesh Deshannavar

Department of Chemical Engineering,
KLE Dr. M.S. Sheshgiri College of Engineering and Technology,
India,
Orchid ID: https://orcid.org/0000-0001-5490-4468

E-mail: deshannavar@gmail.com

Abstract.

The effectiveness of the aviation community depends on accurate forecasting of a 4D aircraft's trajectory, whether in real time or for counter-reality analysis. creating an effective tree-matching technique for the first time in this research to create feature maps that resemble images for historical flight trajectories using high-fidelity meteorological information, including wind, temperature, and convective conditions. Approach the orbit's tracking points as a conditional Gaussian mixture with parameters so they can benefit from our suggested integrated iterative neural network depth generation model. A network of mixed density LSTM decoders and a long memory (LSTM) encoder network make up the terminal. The decoder network learns additional spatial correlations-time from past flight routes and outputs the parameters of the Gaussian composite after the encoder network combines the most recent recorded flight plan information into fixed-length state variables. To learn feature representations from three-dimensional weather feature maps, transformation layers are added into the pipeline.

Keywords: Convolutional Neural Networks, Recurrent neural networks, Gaussian mixture, Weather future prediction.

Journal of Airline Operations and Aviation Management Volume 1 Issue 1
Received Date: 15 May 2022
Accepted Date: 11 June 2022
Published Date: 25 July 2022

1. Introduction

A Traffic Management Initiatives (TMI) are methods for balancing capacity and demand in NAS. TMIs that are properly planned out and implemented are a crucial component of the air traffic control system. These programmes aid in the safe and efficient flow of air traffic. Customers are impacted by all TMI. Radar is used by air traffic controllers to track the location of aircraft in their assigned area, while walkie-talkies are used to communicate with pilots. ATC enforces traffic separation regulations, which guarantee that each aircraft maintains a minimum amount of free space around it at all times, to prevent crashes [1]. Today's air traffic controllers are supported by a range of radar equipment, including Surface Movement Radar (SMR) for field traffic and Primary Surveillance Radar (PSR), Secondary Surveillance Radar (SSR), and Mode S for air traffic monitoring. Specific ATC roles include ground controller, local controller, traffic customs, and flight data, according to the FAA controller staffing plan (FAA pdf source). Finding out the weather in a specific place is crucial since it frequently leads to unneeded flying hassles. There are several ways that weather might affect a flight [2].

It is crucial for pilots to comprehend the weather and strive toward forecasting it since it might affect the choices they must make[11]. If attention is not paid to it, the weather might easily mean the difference between a safe and easy flight and a deadly one [4]. In an effort to cover the whole United States at all altitudes, the Aviation Weather Center strives to collect weather data given by NOAA and present it in a number of forms. Pilots frequently use this service, regardless of the type of aircraft they fly[3].

One advantage of an aeronautical weather centre is that it enables pilots to examine the local weather in the area of the airport, which may be helpful for obtaining weather information for take-off and landing[5]. In order to evaluate flight paths, pilots will also be able to observe weather across the US.

Both text and visual graphics that present all pertinent information may be used to convey this information. All of this weather data will be read, decoded, and interpreted by aspiring pilots.

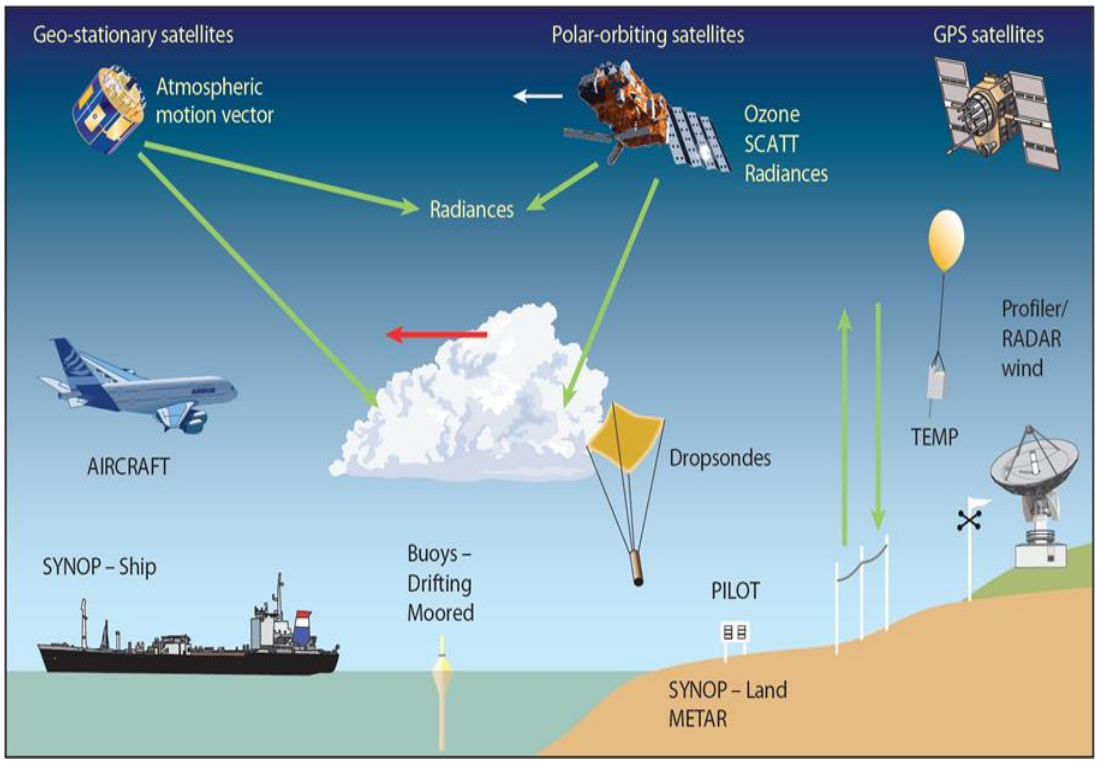

Figure 1: Weather reports for aircraft[7]

2. METHODOLOGY OF DEEP GENERATIVE NEURAL NETWORKS:

A pair of neural networks are pitted against one another to create the GAN. The transmitter network is one of the pair, while the discriminator network is the other. The discriminator determines if the photos are real or false and adds the true images to test. The generator trains the model repeatedly until a divine image is created. Which pictures originate from the transmitter and which from the real are distinguished by the discriminator. A fresh picture is generated automatically by the generator. The generator is taught to distinguish between fake and real pictures.

Before constructing the discriminator, it is necessary to comprehend the problem's condition, organise the pixel logic for the neural network, classify the general condition for a typical image, and ensure that the logic applies to all conditions. The discriminator's output is the result of sigmoid activation.

The neural network is reversed to apply the initial condition, the bias is placed in the same location, the values are based on the same condition, and all the neurons in the output layer are activated using an activation function.

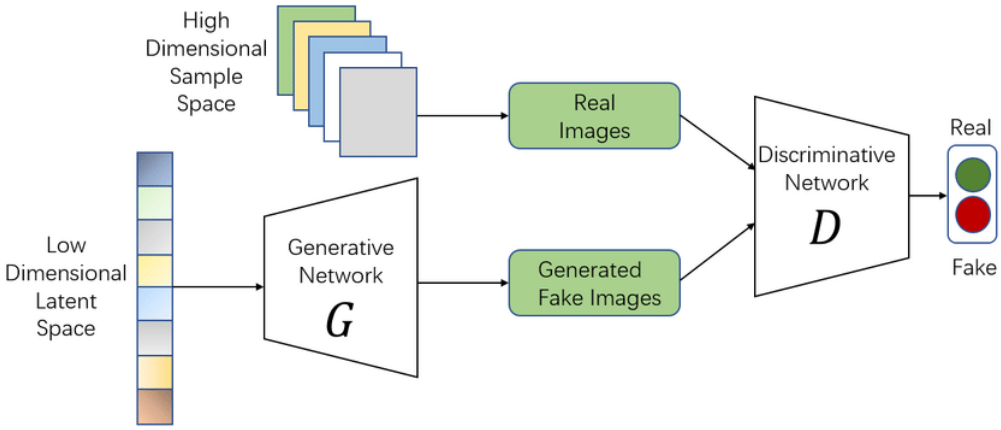

Figure 2: Generative Adversarial networks[8]

The loss function for the training procedure is the log loss function. The log attenuation indicates how likely it is for each prediction to come true. As a result, increasing probability is comparable to decreasing the mean square error (MSE), which is why regression issues frequently utilise this error function.

$$-\begin{cases} \log a_i, & y_i = 1, \\ \log(1-a_i), & y_i = 0. \end{cases}$$

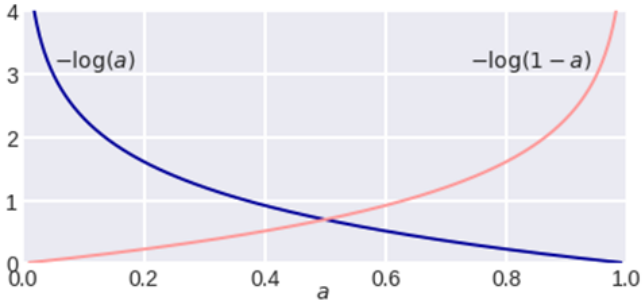

Figure 3: log loss function

3. PREPERATION OF DATA FOR PLANNING THE FLIGHT PATH

Longitude, latitude, time, and altitude are the four datasets that must be used to design the flight route of an aeroplane in four dimensions. One minute, one minute, one minute, and 100 feet is the resolution of those measurements. the collection of flight tracks data that the traffic flow management system provides (TFMS)[9]. The dataset of the flight plan, which comprises the 2-dimensional plain coordinates of longitude and latitude for each trip, is the other piece of information that was taken. The final data, which includes information about the atmosphere, such as wind speed and air temperature, is derived from sections on weather forecasts. This information is derived from flights at two airports.

	LONGITUDE	LATITUDE	FLT_PLAN_ID
0	-95.333333	29.983333	FP_00001
1	-92.750000	33.250000	FP_00001
2	-89.983333	35.016667	FP_00001
3	-84.050000	36.950000	FP_00001
4	-81.116667	37.783333	FP_00001

Figure 4: flight plan data

	Unnamed: 0	FID	Elap_Time	Lat	Lon	Alt	DT	Speed	Elap_Time_Diff	course
0	86898	20130801438231	2013-08-01 00:36:00	29.950000	-95.333333	10	0.0	0.060165	18318960.0	1.850043
1	86899	20130801438231	2013-08-01 00:37:00	29.933333	-95.266667	29	60.0	0.088314	18319020.0	1.570071
2	86900	20130801438231	2013-08-01 00:39:00	29.933333	-95.100000	50	120.0	0.074158	18319140.0	1.570071
3	86901	20130801438231	2013-08-01 00:41:00	29.933333	-94.933333	84	120.0	0.116829	18319260.0	1.651599
4	86902	20130801438231	2013-08-01 00:43:00	29.916667	-94.700000	123	120.0	0.102578	18319380.0	1.732296

Figure 5: Flight track data

The clearance of the path in the air is contained in the flight traffic data[10]. The cleared route may be calculated using data from TFMS. The weather information is obtained from weather report centres, and from there, planes can choose to land at certain stations or, in the event that rain or fog pose a threat to the aircraft, these data will also be required for forecasting and planning the flight path. The wind field data is also necessary since the wind speed indicates whether or not there are wind cyclones or other dangers in the vicinity[12].

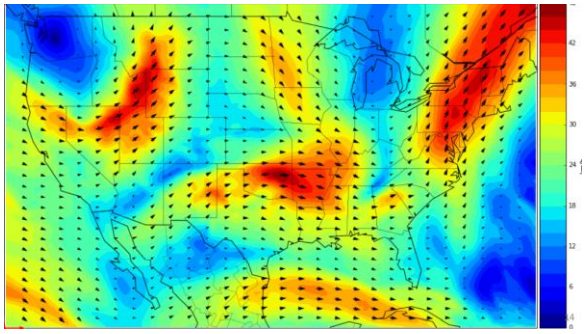

Figure 6: Wind field diagram

The flights of spatial and temporal discontinuities are removed from the data during pre-processing. then by removing one of every two track points during the downsampling flight. The data collection includes information from 1679 flights. data scaling using a common scalar. This scaling transforms the data into a range between 0 and 1. The tables below display four data.

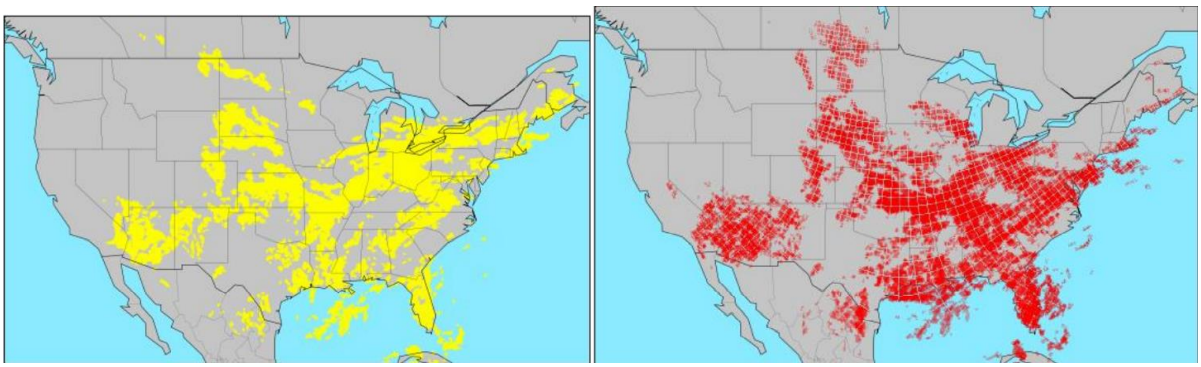

Figure 7: Weather data

Yellow dots on the map indicate the beginning data, which describes the perimeter and height of the convective weather polygon. The other picture depicts arbitrary meteorological data, which will be recorded as binary variables in the matrix covered by our georeferencing grid. The graph's red dots represent items that are non-zero.

3.1.Feature extraction:

The plane's trajectory is dotted. Create a 20x20 grid matrix in a 2D grid with one side centred on the line's point and the other side oriented by azimuth for each track point. Buffers for altitude and time are used. The lowest terrain or obstruction in a given location is used to compute the minimum altitude, with an added margin for error.

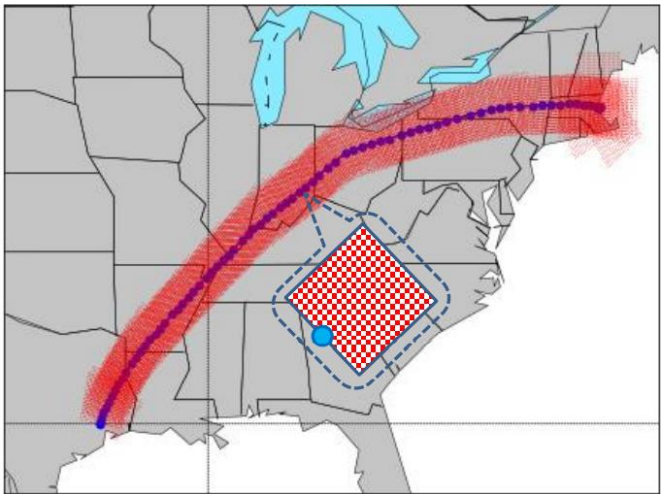

Figure 8: Flight grid path

3.2. TREE MATCHING TECHNIQUE

Tree matching issues pop up in many different computing contexts. That is, after two nodes are matched, matches must also be made for their descendants. Flexible tree matching that has learnable parameters from structured prediction methods. In many situations when the hierarchy is suggestive rather than deterministic, this approach can be helpful for matching. The tree matching approach is utilised in this challenge to identify the flight paths that are closest in time. to determine which location instances are most near the flight grid path.

3.3. Convolutional neural network:

The decoder employs a convolutional neural network. The block data includes the input features as fitting features. Then, with step 2, a complicated layer with the filter applied is 6x6x16 in size, while step 1's other two layers are 8x8x16 and 3x3x32 in size. Due to the information provided by the weather report, avoid grouping and padding procedures. Every detail is significant. 32 neurons operate as feature representations from high dimensional weather feature blocks in the output's dense layer.

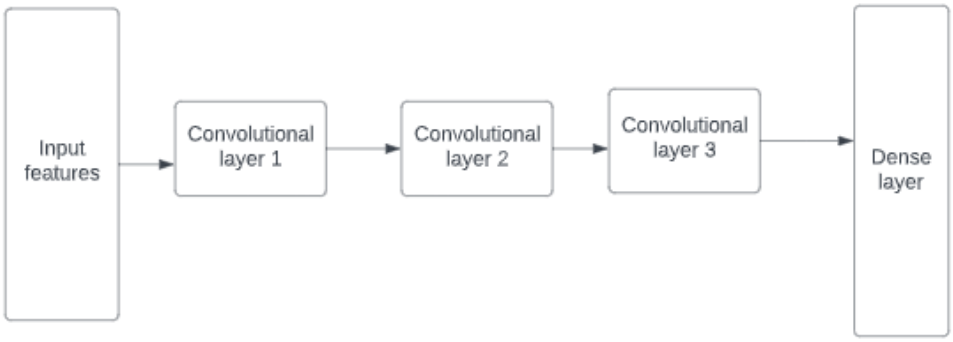

Figure 9: Convolutional neural network (CNN)

3.4. ENCODER DECODER LSTM:

Two models make up this architecture: one reads the input string and encodes it into a fixed-length vector, and the other decodes the fixed-length vector to produce the anticipated sequence. The architecture is known as an LSTM Encoder-Decoder created particularly for seq2seq challenges because to the utilisation of concert patterns. For encoder networks, the flight plan is initially communicated to the LSTM network, followed by the flight route and the relevant characteristic being broadcast to the decoder network, loss, and composite neural network. feature is utilised, as demonstrated in the design, in the LSTM network.

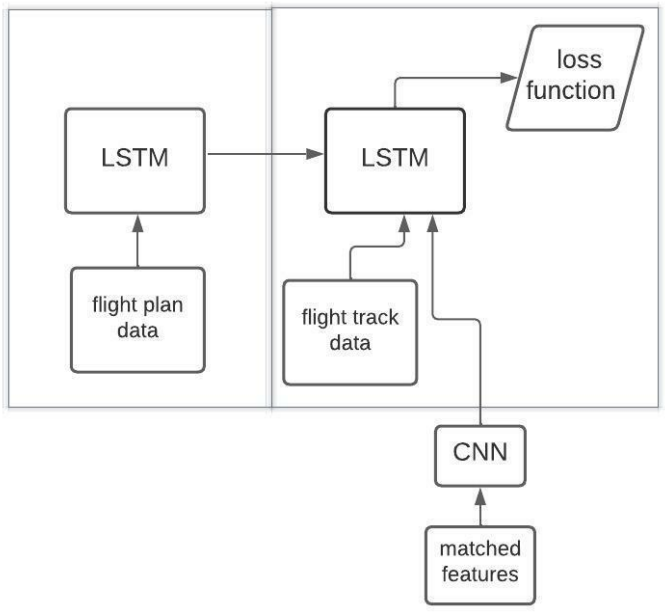

Figure 10: Encoder decoder LSTM

4. RESULTS AND DISCUSSIONS

In this sequence-to-sequence model, the raw features from the wind, weather, and convection data are used as input for the convolutional neural networks, and the visualisation graph is produced by training the model with the log loss loss function.

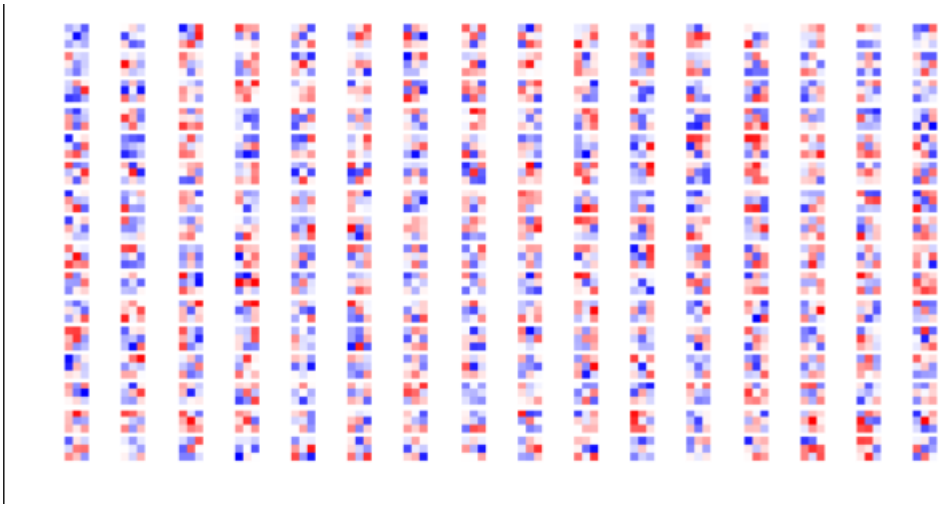

Figure 11: Raw features (Input data)

```
import numpy as np
import tensorflow as tf
import os
from visualize_graph import visual_graph, visualize_raw_weights
import pickle

feature_cubes_pointer = np.load('../data/processed_data/feature_cubes.npz')
feature_cubes = feature_cubes_pointer['feature_cubes']
std_arr_pointer = np.load('../data/processed_data/standardize_arr.npz')
feature_mean = std_arr_pointer['feature_mean']
feature_std = std_arr_pointer['feature_std']
feature_cubes_feed = (feature_cubes - feature_mean)/feature_std

#graph
restored_model_path = 'visual_network/model.ckpt-99'
config_path = 'configs/encoder_decoder_nn.ini'
visual_graph_class = visual_graph(config_path, restored_model_path)

idx = np.random.randint(0, feature_cubes_feed.shape[0], size = 2)
sample_feature_cubes = feature_cubes_feed[idx, None, :, :, :]
print(sample_feature_cubes.shape)

conv1_out, conv2_out, conv3_out = visual_graph_class.feed_fwd_convlayer(feed_input=sample_feature_cubes)
with open('storage.pkl', 'wb') as pfile:
    pickle.dump(pfile, (conv1_out, conv2_out, conv3_out))

weights = visual_graph_class.weights
visualize_raw_weights(weight_var=weights['wc1:0'], fig_size = (8, 2))
visualize_raw_weights(weight_var=weights['wc2:0'], fig_size = (8,4))
visualize_raw_weights(weight_var=weights['wc3:0'], fig_size = (8,4))
plt.show()
```

1679 flights are included in the preprocessed dataset, which is split into two sets, with 80 percent being included in the training set and the remaining in the evaluation set. The remainder of the flight trajectory will be predicted by the algorithm using the first 20 real orbital locations and their accompanying feature blocks for each trip on the evaluation set as a sequence of observations. Two instances of our sampling trajectories are shown in orbiting figures, where the red curve represents the most recent flight plan, the green curve the 20 first observed flight pathways, the magenta curve the 19 anticipated flight trajectory, and the blue dotted curve the actual flight path (truth on the ground). Warmer hues

(moving toward red) indicate greater temperatures in the figure's background colours, which reflect typical air temperatures. The red polygons depict convective weather zones, while the arrows show wind direction and speed. The anticipated flight path is depicted by the green band, which also shows the three standard errors that make up the 99.7% confidence interval. In the illustration, the right subplot features intense convection whereas the left subplot displays an example of modest convection activity. With a little divergence in the middle of the orbit, the two projected trajectories mostly match the actual course taken (truth on the ground). The prediction intervals, which are shown by the histogram's slender green bars, however, cover these variations. The produced test picture samples and their projected path coordinates with time t are shown in the following figure.

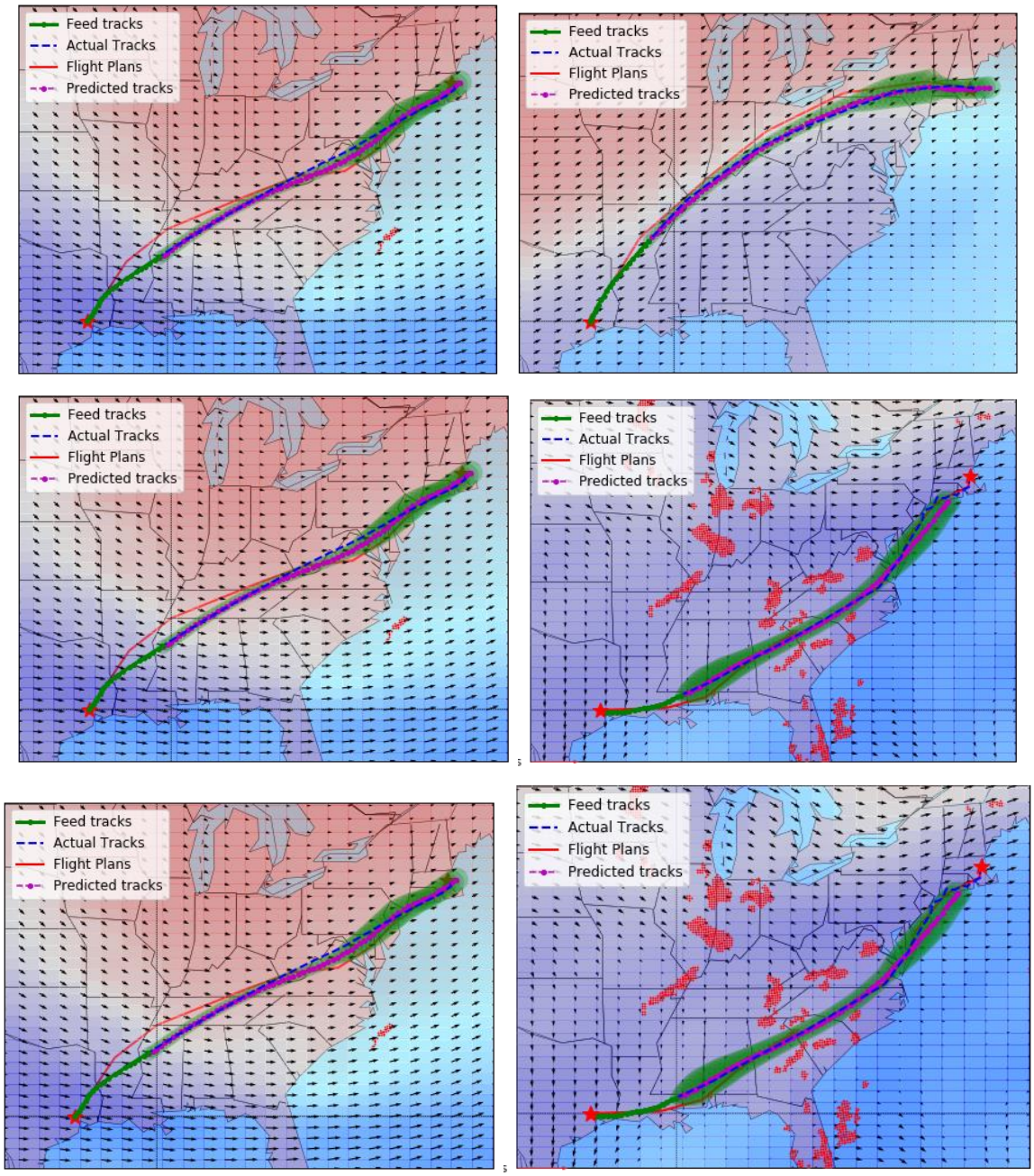

Figure 12: predicted flight trajectory path

5. CONCLUSION:

This paper uses the most recent saved flight plan and height weather information to estimate actual 4D aircraft trajectory. Our method uses a matching algorithm, a worm generation model, a training framework, and an inference framework to resolve this "sequence-to-sequence" problem. While 337 flights were utilised for evaluation, 1342 flights were used for training. While visualising the composite classes,the learnt filters effectively distinguish between convective weather and weather-related variables. Prediction error is measured using four measures. The mean absolute vertical error is around 2800 feet, and the mean absolute lateral error of the point and route is approximately 50 knots. But also detect significant forecast errors for flights (outliers) with irregular departure processes, which will investigate in the future to do a search. Future work will also involve adding new characteristics to our matching algorithm, such as neighbouring air traffic management activities (runway miles, airspace flow timetables, etc.).

References

[1] J. Kaneshige, J. Benavides, S. Sharma, L. Martin, R. Panda and M. Steglinski, "Implementation of a trajectory prediction function for trajectory based operations," in AIAA Aviation Atmospheric Flight Mechanics Conference, No. AIAA, Vol. 2198, 2014.

[2] A. de Leege, M. Van Paassen and M. Mulder, "A machine learning approach to trajectory prediction," in AIAA GNC Conference and Exhibit, Boston, MA, 2013.

[3] P. Y. Choi and M. Hebert. Learning and predicting moving object trajectory: a piecewise trajectory segment approach. Technical report, Carnegie Mellon University School of Computer Science, Pittsburgh, PA, August 2006.

[4] S. Ayhan and H. Samet, "Aircraft Trajectory Prediction Made Easy with Predictive Analytics," in 22nd International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, 2016.

[5] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural computation, vol. 9, no. 8, pp. 1735-1780, 1997.

[6] A. Graves and N. Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In Proceedings of the 31st International Conference on Machine Learning (ICML-14), pages 1764–1772, 2014.

[7] "Observations | ECMWF," ECMWF, Nov. 29, 2013. https://www.ecmwf.int/en/research/data-assimilation/observations (accessed Jul. 01, 2022).

[8] Cai, Likun & Chen, Yanjie & Cai, Ning & Cheng, Wei & Wang, Hao. (2020). Utilizing Amari-Alpha Divergence to Stabilize the Training of Generative Adversarial Networks. Entropy. 22. 410. 10.3390/e22040410.

[9] J. Chorowski, D. Bahdanau, K. Cho, and Y. Bengio. Endto-end continuous speech recognition using attentionbased recurrent nn: First results. arXiv preprint arXiv:1412.1602, 2014.

[10] J. Chung, K. Kastner, L. Dinh, K. Goel, A. C. Courville, and Y. Bengio. A recurrent latent variable model for sequential data. CoRR, abs/1506.02216, 2015.

[11] I. Sutskever, O. Vinyals, and Q. V. Le. "Sequence to sequence learning with neural networks." In Advances in neural information processing systems, pp. 3104-3112. 2014.

[12] A. Karpathy, A. Joulin, and F. Li "Deep fragment embeddings for bidirectional image sentence mapping." In Advances in neural information processing systems, pp. 1889-1897. 2014.