aerospace

MDPI

Communication

Air Traffic Prediction as a Video Prediction Problem Using Convolutional LSTM and Autoencoder

Hyewook Kim [1] and Keumjin Lee [1,2,*]

[1] Department of Air Transportation and Logistics, Korea Aerospace University, Deogyang-gu, Goyang-si 10540, Gyeonggi-do, Korea; urahh123@hanmail.net

[2] Department of Smart Air Mobility, Korea Aerospace University, Deogyang-gu, Goyang-si 10540, Gyeonggi-do, Korea

* Correspondence: keumjin.lee@kau.ac.kr

Abstract: Accurate prediction of future air traffic situations is an essential task in many applications in air traffic management. This paper presents a new framework for predicting air traffic situations as a sequence of images from a deep learning perspective. An autoencoder with convolutional long short-term memory (ConvLSTM) is used, and a mixed loss function technique is proposed to generate better air traffic images than those obtained by using conventional L_1 or L_2 loss function. The feasibility of the proposed approach is demonstrated with real air traffic data.

Keywords: air traffic prediction; trajectory prediction; deep learning; video prediction

check for updates

Citation: Kim, H.; Lee, K. Air Traffic Prediction as a Video Prediction Problem Using Convolutional LSTM and Autoencoder. *Aerospace* **2021**, *8*, 301. https://doi.org/10.3390/aerospace8100301

Academic Editor: Alexei Sharpanskykh

Received: 26 August 2021
Accepted: 11 October 2021
Published: 14 October 2021

1. Introduction

Air traffic prediction over different time horizons is one of the important research topics in the field of air traffic management for optimizing the use of airspace and airport resources in addition to safety purposes, such as conflict probes [1,2]. Various methods for aircraft trajectory prediction have been proposed [3–7], but their accuracy is still limited by many factors [8]. One of the major factors that contribute to an inaccurate trajectory prediction is that the trajectories often deviate from the filed flight plans [9]. Various methods have been proposed to infer the true paths of flights, but it is still challenging to effectively incorporate the interacting dynamics of multiple aircraft [10–13]. A more holistic approach might be needed to consider complex interconnections between trajectories.

Recently, several attempts have been made to consider the overall air traffic situation as image data in a specific airspace. One study proposed a method to detect an anomaly in ADS-B messages [14]. An air traffic image generated from a given ADS-B message was reproduced by an autoencoder model, and the prediction error was then used to determine the authenticity of the messages. In another study, a convolutional neural network (CNN) was used to extract important features from a multichannel air traffic scenario image to evaluate its complexity [15].

Building upon these previous studies, this paper introduces a new paradigm for air traffic prediction. In conventional approaches, the aircraft trajectories are individually predicted and combined to construct a future traffic situation. However, in the proposed method, the overall air traffic situation in a specific airspace at an instantaneous time is considered as an image, and a data-driven model is trained to learn how this image changes over time as illustrated in Figure 1. In other words, the proposed method considers the task of air traffic prediction as a video prediction problem from a deep learning perspective. Once the model is trained with historical data, future air traffic situations can be predicted as a sequence of subsequent images based on the model. Note that a sequence of air traffic images contains a wealth of information about not only individual aircraft trajectories, but also their interactions with neighboring traffic and other environmental factors, such as the runway configuration.

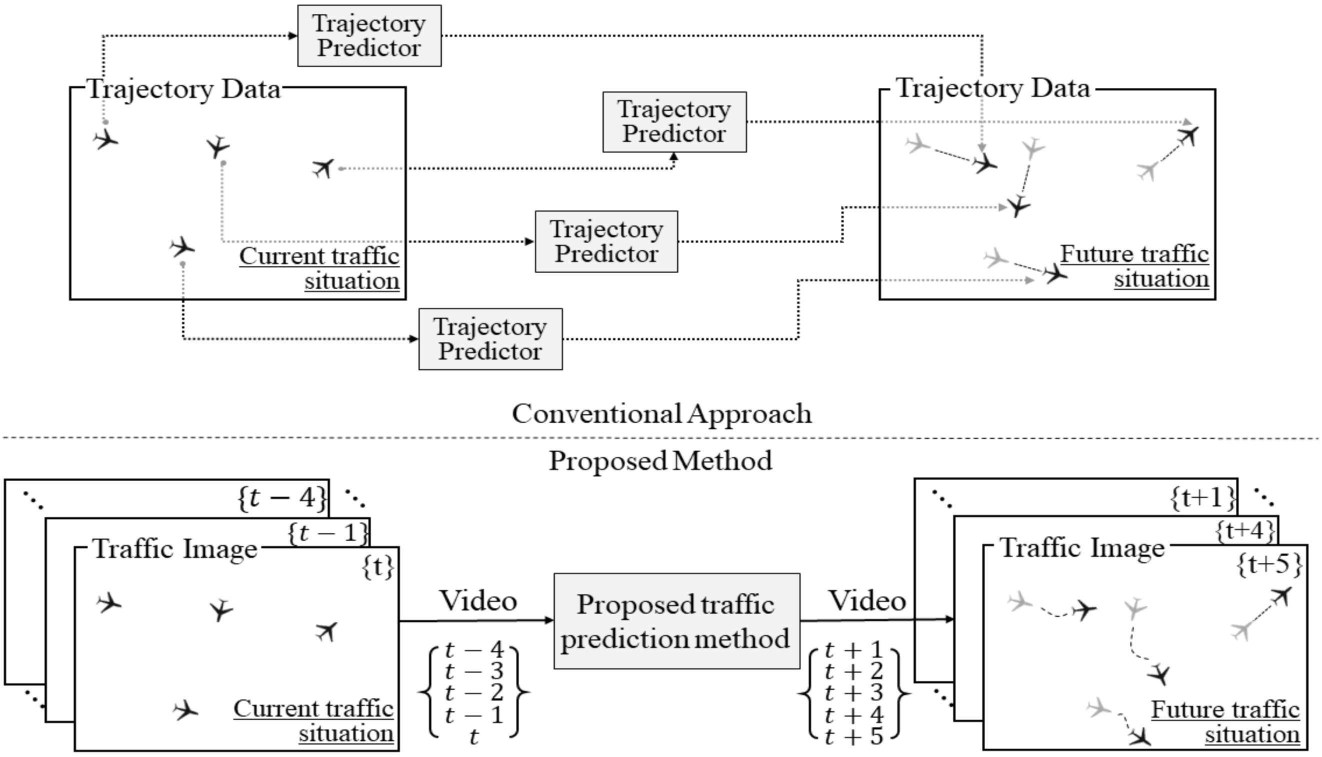


Figure 1. Traffic prediction as a video prediction problem; (**top**) conventional approach; (**bottom**) proposed approach.

The remainder of this communication is organized as follows. Section 2 describes the analytical framework of the proposed approach for air traffic prediction, and Section 3 demonstrates the feasibility of the proposed approach with real air traffic data. Finally, concluding remarks and future works are presented in Section 4.

2. Methodology

Let $X_t \in R^{w \times h \times c}$ denote the air traffic situation at time t as an image for which w, h, and c are the width, height, and channel size of the image, respectively. The proposed problem of air traffic prediction involves generating the next K air traffic images $(X_{t+1}, X_{t+2} \cdots X_{t+K})$ with maximum likelihood L given the previous J air traffic images $(X_{t-J+1}, X_{t-J+2}, \cdots X_t)$:

$$\hat{X}_{t+1}, \ldots, \hat{X}_{t+K} = \underset{X_{t+1},\ldots,X_{t+K}}{argmax} \ L\left(X_{t+1},\ X_{t+2} \ldots, X_{t+K} \,\middle|\, X_{t-J+1}, X_{t-J+2}, \ldots, X_t\right) \quad (1)$$

In this paper, an autoencoder architecture is used to predict future air traffic images. The autoencoder architecture consists of two separate networks: (1) an encoder; and (2) a decoder. The encoder network is used to compress an input sequence $(X_{t-J+1},\ X_{t-J+2}, \ \cdots X_t)$ into embedding vectors, $v's$, while the decoder network acts as a predictor to generate a sequence of the next K image frames $(X_{t+1}, X_{t+2} \ \cdots \ X_{t+K})$ for given embedding vectors, as shown in Figure 2.

To better capture the spatiotemporal correlation in air traffic images, convolutional long short-term memory (ConvLSTM) is used in both the encoder and decoder networks [16]. LSTM has been proven to be stable for handling the temporal features of sequential data, but it easily overlooks the spatial natures of images [17,18]. ConvLSTM is a variant of LSTM for which the input-to-state and state-to-state transitions are exchanged with convolutional operations with an $M \times N$ grid called a kernel (Figure 3a). Therefore, ConvLSTM is better at retaining both the temporal and spatial information of an image than conventional LSTM, in which a two-dimensional (2D) image is flattened into a 1D vector (Figure 3b).

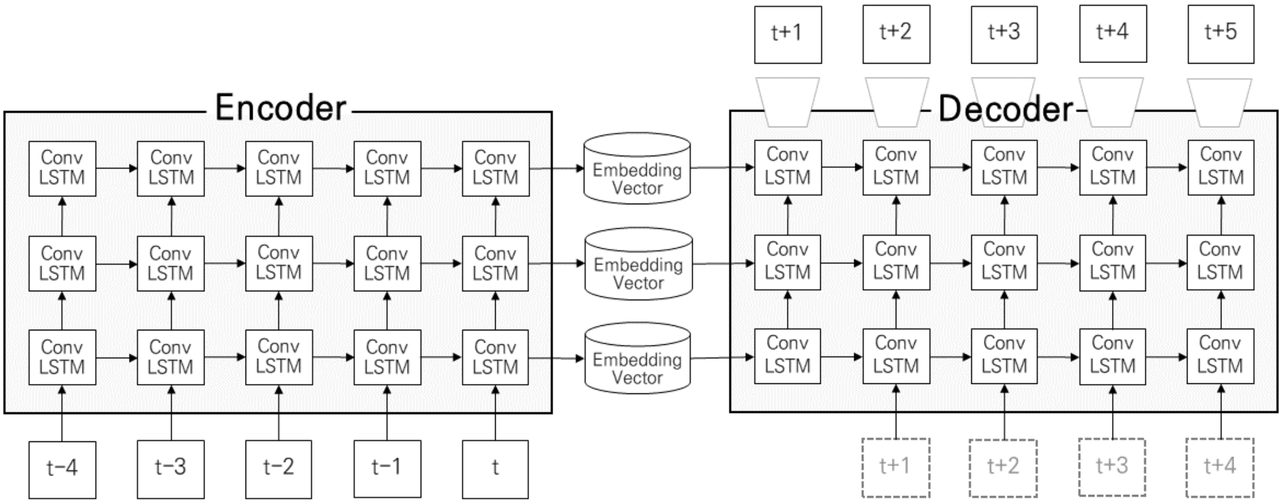


Figure 2. The structure of the convolutional long short-term memory (ConvLSTM) autoencoder.

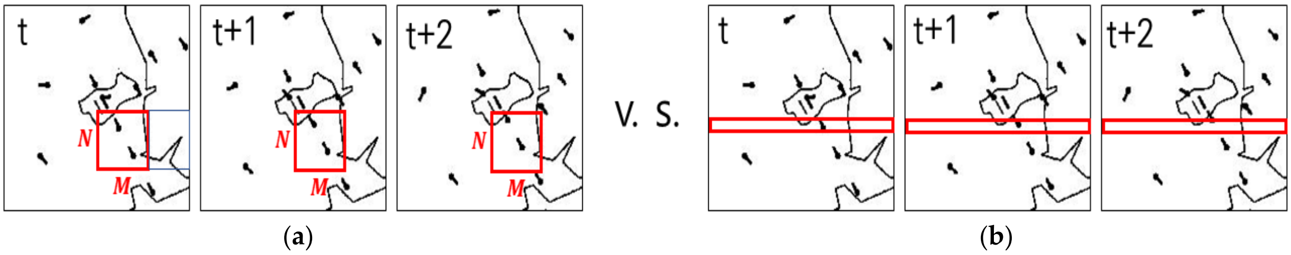


Figure 3. The receptive field (red box) of the network: (**a**) ConvLSTM; (**b**) LSTM.

3. Numerical Example

3.1. Air Traffic Image Data

A dataset of air traffic images was created based on historical surveillance data for three months of flights over 900 square miles of airspace around Incheon International Airport (ICN), South Korea. The runway configurations of ICN are shown in Figure 4. Due to the close proximity to the border with North Korea, the north wind configuration (33/34) is dominant. At the time corresponding to the selected data, three runways were in use, but a fourth runway (the westernmost runway in the figure) began to operate in 2021. Note that this study does not exclude any flight in these three months although the flight patterns could vary especially at night or on the weekend. Segregation of the dataset based on time or weekday would further improve the performances of the proposed method and needs to be explored in future research.

As shown in Figure 5, the time, latitude, longitude, heading, and identification of each flight were extracted every 20 s from the radar surveillance data, and this information was converted into a traffic image combined with the runway configuration data (i.e., location, length, direction) and terrain data [19]. The resolution of the image was chosen to be $164(w) \times 180(h)$ with 1 pixel corresponding to 0.03 square miles of airspace at an instantaneous time. This resolution was the maximum amount that we could apply in this study due to the limited computational power available, and it was ensured that the corresponding size of real airspace for each pixel was at least small enough not to contain more than one flight.

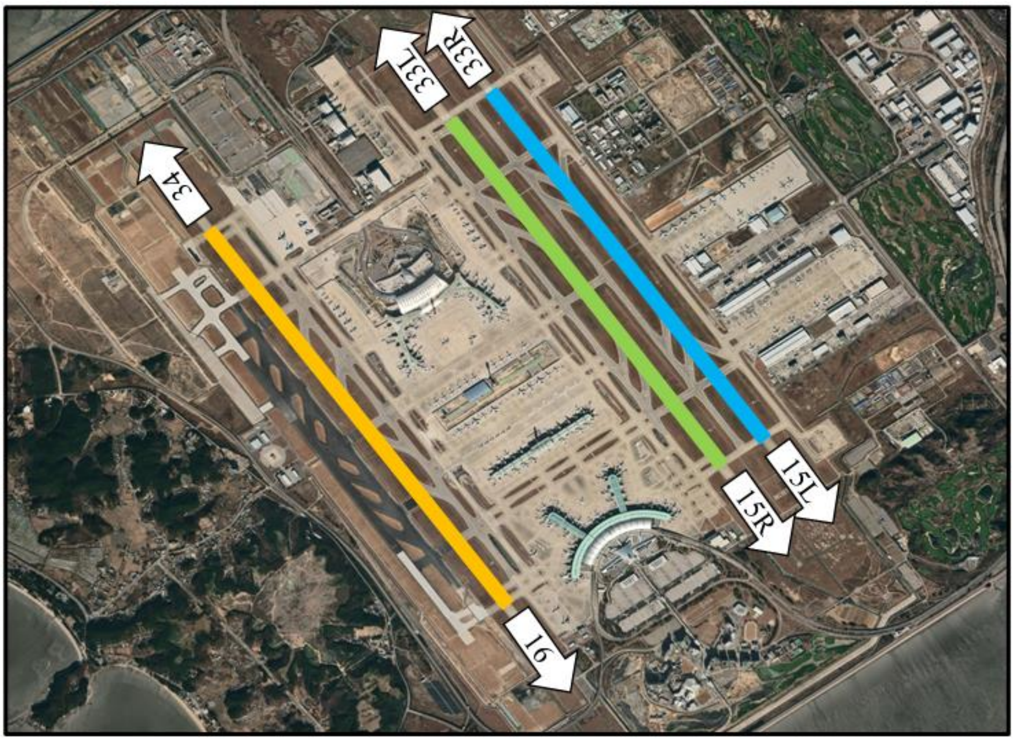
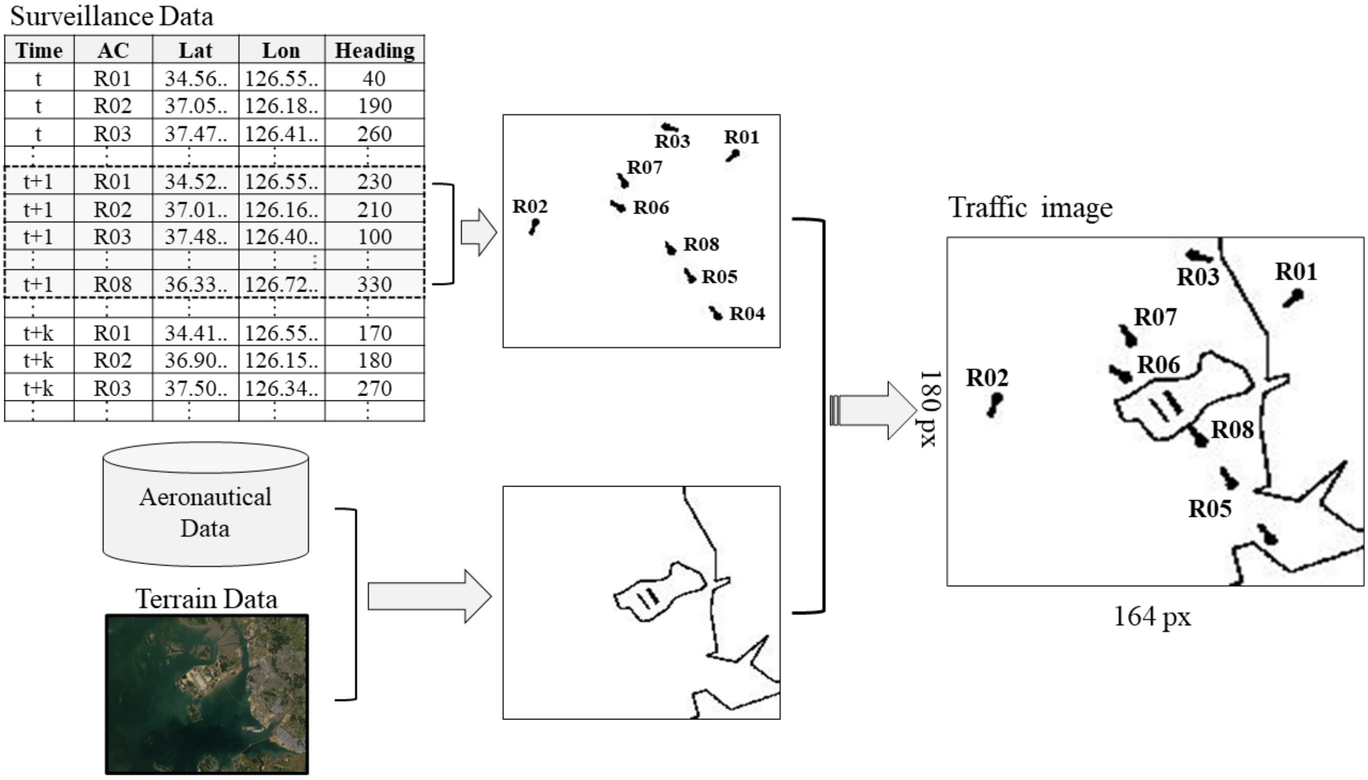


Figure 4. Incheon airport (ICN) runway configuration [20].



Time	AC	Lat	Lon	Heading
t	R01	34.56..	126.55..	40
t	R02	37.05..	126.18..	190
t	R03	37.47..	126.41..	260
⋮	⋮	⋮	⋮	⋮
t+1	R01	34.52..	126.55..	230
t+1	R02	37.01..	126.16..	210
t+1	R03	37.48..	126.40..	100
⋮	⋮	⋮	⋮	⋮
t+1	R08	36.33..	126.72..	330
⋮	⋮	⋮	⋮	⋮
t+k	R01	34.41..	126.55..	170
t+k	R02	36.90..	126.15..	180
t+k	R03	37.50..	126.34..	270
⋮	⋮	⋮	⋮	⋮

Figure 5. Generating air traffic images from surveillance, aeronautical, and terrain data.

3.2. Constructing the Autoencoder Model

As shown in Figure 2, both the encoder and the decoder contain three layers of ConvLSTM, and each layer contains 64 hidden states with 3 × 3 kernels. The model accepts five image frames as input and predicts the next five image frames as output. An additional output layer (3D CNN) is used to reconstruct an output image sequence after the last layer of ConvLSTM. The specifics of the model are shown in Table 1.

Table 1. Type and output shape of each layer in encoder and decoder networks. The output shape is represented as follows: (# of image frames, width, height, # of hidden states).

Network	Layer Type	Output Shape
Encoder	Input Layer	(5, 164, 180, 3)
	ConvLSTM2D 1 (3 × 3 *kernel*, 1 × 1 *padding*, 1 *stride*)	(5, 164, 180, 64)
	ConvLSTM2D 2 (3 × 3 *kernel*, 1 × 1 *padding*, 1 *stride*)	(5, 164, 180, 64)
	ConvLSTM2D 3 (3 × 3 *kernel*, 1 × 1 *padding*, 1 *stride*)	(5, 164, 180, 64)
Decoder	ConvLSTM2D 1 (3 × 3 *kernel*, 1 × 1 *padding*, 1 *stride*)	(5, 164, 180, 64)
	ConvLSTM2D 2 (3 × 3 *kernel*, 1 × 1 *padding*, 1 *stride*)	(5, 164, 180, 64)
	ConvLSTM2D 3 (3 × 3 *kernel*, 1 × 1 *padding*, 1 *stride*)	(5, 164, 180, 64)
	3D CNN (3 × 3 *kernel*, 1 × 1 *padding*, 1 *stride*, *tanh activation*)	(5, 164, 180, 3)

For video prediction problems, two types of loss functions are often used: (1) $L2$ and (2) $L1$ [21]. As shown in Equations (2) and (3), the $L2$ loss function involves minimizing the sum of the squares of differences between the RGB values of the true and the predicted images. In contrast, the $L1$ loss function involves minimizing the absolute differences. In the equations, $y(i,j)$ represents the RGB value of the pixel (i,j).

$$Mean\ Squared\ Error\ (L2) = mean(\sum_{i=1}^{w}\sum_{j=1}^{h}\left(y_{true(i,j)} - y_{predict(i,j)}\right)^2) \tag{2}$$

$$Mean\ Absolute\ Error\ (L1) = mean(\sum_{i=1}^{w}\sum_{j=1}^{h}\left|y_{true(i,j)} - y_{predict(i,j)}\right|) \tag{3}$$

The $L2$ loss function could suffer from blurriness of the output images, while one with $L1$ could suffer from the vanishing of dynamic objects [22,23]. To mitigate these problems, we propose a mixed loss function technique called "*L2 and then L1*". The model was trained with the $L2$ loss function until one-third of the total training steps (i.e., 150 epochs) was reached, and the $L1$ loss function was then used for the last quarter of the training steps (i.e., 50 epochs). Note that the total number of epochs in this study was fixed to be 200. The idea behind the mixed loss function is that pretraining the model with $L2$ could be more effective for capturing the variability in traffic patterns before being trained with $L1$. Similar techniques, such as using a pretrained model or switching to a different loss function during the training process, are often used to train a deep learning model better [24,25]. Although more theoretical justification of the proposed mixed loss function should be explored in future research, it turned out to be quite effective and allowed the model to suffer less from flights vanishing in the predicted images when compared with the case with only $L1$ loss function, as discussed in the next section.

The model was trained with 5000 image sequences using the Adam optimizer [26] and a learning rate starting at 10^{-3}. With a batch size of 16, the training took approximately 60 h using an NVIDIA RTX 3070 (8 GB memory) with a Pytorch/Pytorch Lightning backend [27,28]. An extra 2000 images were used as a validation set to evaluate different choices of the number of layers of the autoencoder network, the kernel size of ConvLSTM, and the type of activation function of the output CNN in terms of the total computation time, the required memory size, and the quality of predicted images.

3.3. Results and Discussion

Since the output of the proposed air traffic prediction model is given in the form of images, a proper definition of how to measure its prediction accuracy is needed for quantitative validation. A per-pixel loss used in the training process is not a good indication of the quality of trajectory prediction, and a different evaluation metric, which probably requires converting the predicted images back to the trajectory data (latitude, longitude, heading, and identification), would be necessary. To that end, extensive experimental work and trade-off analysis are usually required [29], and this type of work is beyond the scope of this paper. At the current stage of this study, a prompt introduction of the proposed approach is the focus. Therefore, only visual evaluation of the predicted images is presented in this paper. Future work should be followed for a more quantitative and exhaustive validation of the proposed approach.

The experimental results from one selected test sample are shown in Figure 6, which shows a series of air traffic images from every 20 s. Figure 6a shows the model's input images, and Figure 6b–e show the future air traffic images predicted by the proposed model with different loss functions ($L2$, $L1$, and $L2$ *and then* $L1$). Figure 6b shows how the aircraft actually flew (i.e., ground truth). Among other test samples, this example was chosen because it contained a mix of landing and departure traffics in different directions. In fact, the model's prediction performances varied depending on the geographical locations and flight directions of traffics, and the air traffic scenario in Figure 6 was considered a good example to illustrate both strengths and weaknesses of the proposed model.

As indicated by the red box in the figure, the images generated with $L2$ suffer from blurriness. In contrast, the images with $L1$ show more vivid trajectories, but some flights gradually vanish as shown in the blue dotted boxes in Figure 6d. However, the model with the mixed loss function generates more vivid and long-lasting trajectories than the models with $L1$ or $L2$ loss functions.

The model is capable of predicting the trajectories of landing/departing traffic to follow the standard arrival routes (STARs) and the standard instrument departures (SIDs) even though such information was not included in the training process. For instance, the landing traffics on the runway 33 (AC1–AC4) in the predicted image (Figure 6e) could stay on their STAR and maintain separations between successive aircraft. Note that the training data contained the flight vectored off from SIDs or STARs; therefore, it might not be too easy for the model to keep the traffic images on SIDs and STARs as illustrated by many blips in the predicted images with the $L2$ loss function in Figure 6c.

It is interesting to note that the model could show that AC1 disappeared from the predicted images because it has landed in scene #5. The proposed model could also make successful predictions for departing flights (such as AC5–AC7) from the same runway where they make a U-turn to the right along the SID when they arrived 10.6 miles away from the runway threshold, as shown in Figure 6e. It should also be noted that the proposed model allows for a new aircraft to start to appear in the predicted image (AC10 in Figure 6e #9) although that aircraft did not exist in the input images. The proposed model still suffers from the loss of some flights in the predicted images. For instance, AC8 departed from runway 34 of ICN and AC9 departed from the Gimpo Airport (GMP) nearby and they vanish more quickly from the predicted images than other flights.

The performance of the proposed approach for air traffic prediction might not yet be sufficient for practical applications. However, its performance could be significantly improved by exploring other state-of-the-art techniques for video prediction problems and by collecting more air traffic data. Importantly, this new approach could play a role in overcoming one of the main hurdles in air traffic prediction, namely, incorporating simultaneous and multiple interactions between neighboring aircraft. Work remains to be done to further investigate the potential and limitations of the proposed method in addition to how it can be used in a variety of applied contexts. The proposed approach might not be accurate enough to be used as a standalone application, but it could be used as a hybrid form together with the conventional trajectory prediction methods.

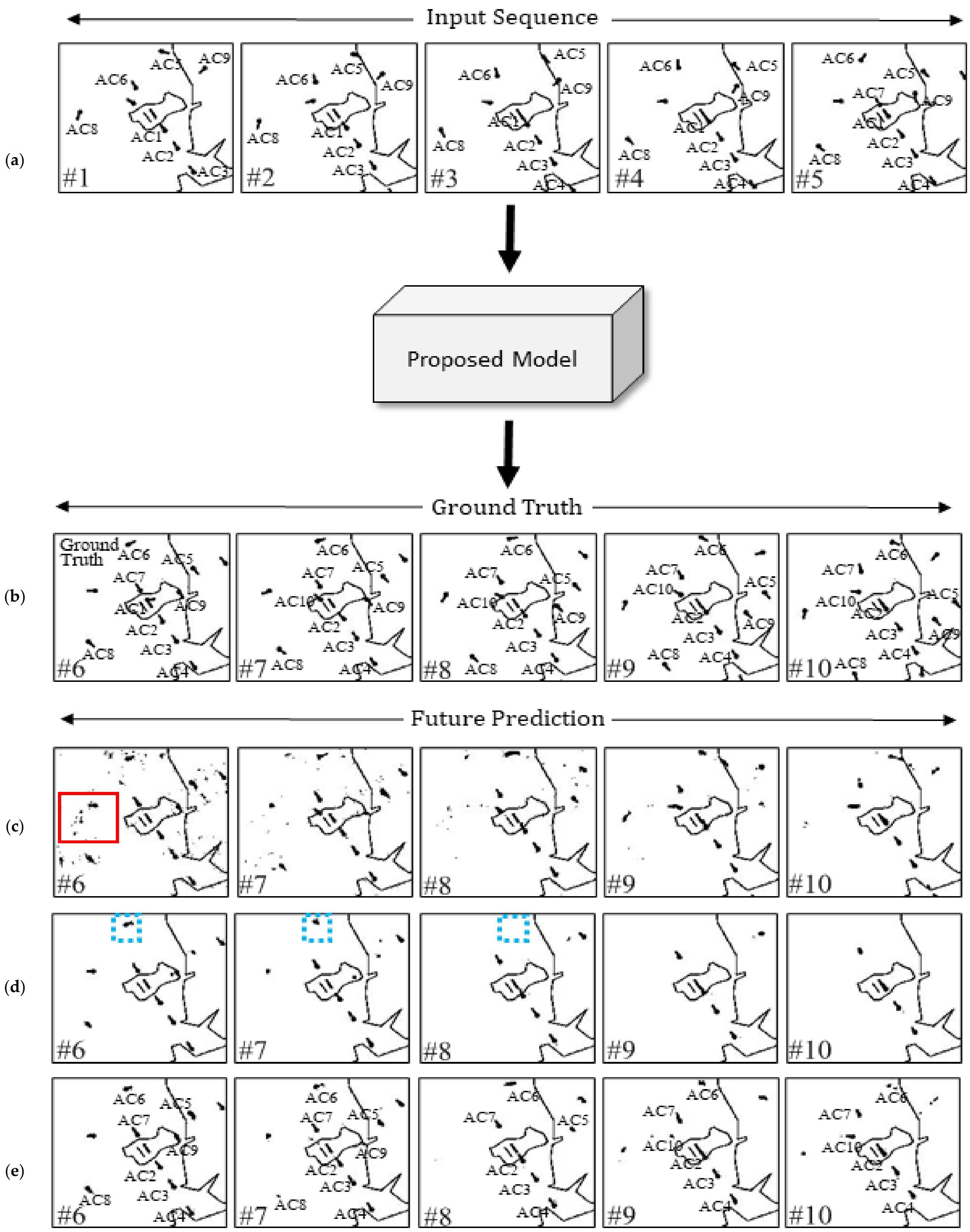


Figure 6. (**a**,**b**) Ground truth traffic images (#1–#10) recorded at 20 s intervals, and (**c**–**e**) predicted frames (#6–#10) of the model based on each loss function: (**c**) *L*2; (**d**) *L*1; (**e**) *L*2 and then *L*1.

4. Conclusions

A new framework was proposed to predict an entire air traffic scene from a deep learning perspective, unlike a conventional approach to trajectory prediction, where trajectories of individual aircraft are the focus. A ConvLSTM-based autoencoder architecture was used, and three different loss functions were compared. The main purpose of this initial study is to show the feasibility of this new approach for air traffic predictions, and many future works are necessary.

For future works, the proposed method should be more extensively developed. Particularly, a method is needed to incorporate the altitude information of aircraft into a traffic image. Altitude information is important from an air traffic management perspective, and its inclusion in the traffic image would broaden the applicability of the proposed method. Additionally, other data-driven techniques that may better fit the proposed problem of air traffic prediction should be explored, and the associated hyperparameters should be further calibrated. A new metric is also needed to evaluate the performance of predicted traffic images more quantitatively and comprehensively for a broad range of air traffic scenarios. Although this study focused on presenting the proposed approach, work still remains to identify its various use cases.

Author Contributions: Conceptualization, H.K. and K.L.; methodology, H.K. and K.L.; software, H.K.; validation, H.K. and K.L.; formal analysis, H.K. and K.L.; investigation, H.K. and K.L.; data curation, H.K. and K.L.; writing—original draft preparation, H.K. and K.L.; writing—review and editing, H.K. and K.L.; supervision, K.L.; funding acquisition, K.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Ministry of Land, Infrastructure, and Transport, South Korea, under contract 21DATM-C162722-01.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Due to the nature of this research, participants of this study did not agree for their data to be shared publicly, so supporting data is not available.

Conflicts of Interest: The authors declare that they have no conflict of interest.

References

1. Warren, A. Trajectory prediction concepts for next generation air traffic management. In Proceedings of the 3rd USA/Europe Air Traffic Management R&D Seminar, Napoli, Italy, 13–16 June 2000.
2. Liu, W.; Hwang, I.S. Probabilistic Trajectory Prediction and Conflict Detection for Air Traffic Control. *J. Guid. Control Dyn.* **2011**, *34*, 1779–1789. [CrossRef]
3. Gallo, E.; Lopez-Leones, J.; Vilaplana, M.A.; Navarro, F.A.; Nuic, A. Trajectory computation infrastructure based on BADA aircraft performance model. In Proceedings of the IEEE/AIAA 26th Digital Avionics Systems Conference, Dallas, TX, USA, 21–25 October 2007. [CrossRef]
4. Musialek, B.; Munafo, C.F.; Ryan, H.; Paglione, M. *Literature Survey of Trajectory Predictor Technology*; National Technical Information Service: Springfield, VA, USA, 2010; pp. 1–31.
5. Swierstra, S. *Common Trajectory Predictor Structure and Terminology in Support of SESAR and NextGen*; Federal Aviation Administration: Washington, DC, USA, 2010; pp. 1–25.
6. Chai, H.; Lee, K. En-route arrival time prediction via locally weighted linear regression and interpolation. In Proceedings of the IEEE/AIAA 38th Digital Avionics Systems Conference, San Diego, CA, USA, 8–12 September 2019; pp. 1–6. [CrossRef]
7. Vourous, G. *Data-Driven Aircraft Trajectory Prediction Exploratory Research*; NASA: Washington, DC, USA, 2017; pp. 9–13.
8. Rudnyk, J.; Ellerbroek, J.; Hoekstra, J. Trajectory Prediction Sensitivity Analysis Using Monte Carlo Simulations Based on Inputs' Distributions. *AIAA JAT* **2019**, *27*, 181–198. [CrossRef]
9. Mondoloni, S.; Bayraktutar, I. Impact of Factors, Conditions and Metrics on Trajectory Prediction Accuracy. In Proceedings of the 6th USA/Europe ATM R&D Seminar, Baltimore, MD, USA, 27–30 June 2005.
10. Yepes, J.; Hwang, I.; Rotea, M. New Algorithms for Aircraft Intent Inference and Trajectory Prediction. *AIAA JGCD* **2007**, *30*, 370–382. [CrossRef]
11. Roy, K.; Levy, B.; Tomlin, C.J. Target tracking and Estimated Time of Arrival (ETA) Prediction for Arrival Aircraft. In Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit, Keystone, CO, USA, 21–24 August 2006. [CrossRef]
12. Hong, S.; Lee, K. Trajectory Prediction for Vectored Area Navigation Arrivals. *AIAA JAIS* **2015**, *12*, 490–502. [CrossRef]

13. Jung, S.; Hong, S.; Lee, K. A Data-Driven Air Traffic Sequencing Model Based on Pairwise Preference Learning. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 803–816. [CrossRef]
14. Akerman, S.; Habler, E.; Shabtai, A. VizADS-B: Analyzing Sequences of ADS-B Images Using Explainable Convolutional LSTM Encoder-Decoder to Detect Cyber Attacks. *arXiv* **2019**, arXiv:1906.07921.
15. Xie, H.; Zhang, M.; Ge, J.; Dong, X.; Chen, H. Learning Air Traffic as Images: A Deep Convolutional Neural Network for Airspace Operation Complexity Evaluation. *Complexity* **2021**, *2021*, 6457246. [CrossRef]
16. Shi, X.; Chen, Z.; Wang, H.; Yeung, D.Y.; Wong, W.K.; Woo, W.C. Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting. *NIPS* **2015**, *28*, 802–810.
17. Wang, Y.; Long, M.; Wang, J.; Gao, Z.; Yu, P.S. PredRNN: Recurrent Neural Networks for Predictive Learning using Spatiotemporal LSTMs. *NIPS* **2017**, *30*, 879–888.
18. Jeon, H.; Kum, D.; Jeong, W. Traffic Scene Prediction via Deep Learning: Introduction of Multi-Channel Occupancy Grid Map as a Scene Representation. *IEEE Intell. Veh. Symp. IV* **2018**, *1*, 1496–1501. [CrossRef]
19. Republic of Korea, Office of Civil Aviation (ROK). *Aeronautical Information Publication*; RKSI Aerodrome Chart 2-1; ROK: Seoul, Korea, 2020.
20. Google Earth 7.3. 2021. Incheon International Airport 37°27′45″ N 126°26′21″ E. Available online: https://earth.google.com/web/@37.46021582,126.43471177,1.59991967a,24301.36507931d,34.99998106y,0h,0t,0r (accessed on 10 October 2021).
21. Zhou, Y.; Dong, H.; Saddik, A. Deep Learning in Next-Frame Prediction: A Benchmark Review. *IEEE Access* **2020**, *8*, 69273–69283. [CrossRef]
22. Zhao, H.; Gallo, O.; Frosio, I.; Kautz, J. Loss Functions for Image Restoration with Neural Networks. *IEEE Trans. Comput. Imaging* **2017**, *3*, 47–57. [CrossRef]
23. Shouno, O. Photo-Realistic Video Prediction on Natural Videos of Largely Changing Frames. *arXiv* **2020**, arXiv:2003.08635v1.
24. Zhuang, R.; Qi, Z.; Duan, K.; Xi, D.; Zhu, Y.; Zhu, H.; Xiong, H.; He, Q. A Comprehensive Survey on Transfer Learning. *arXiv* **2019**, arXiv:1911.02685. [CrossRef]
25. Barron, J.T. A General and Adaptive Robust Loss Function. *arXiv* **2017**, arXiv:1701.03077.
26. Kingma, D.P.; BA, J.L. ADAM: A Method For Stochastic Optimization. *arXiv* **2014**, arXiv:1412.6980.
27. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *NIPS* **2019**, *32*, 8024–8035.
28. Pytorch Lightning. Available online: https://github.com/PyTorchLightning/pytorch-lightning (accessed on 10 October 2021).
29. Johnson, J.; Alahi, A.; Fei-Fei, L. Perceptual Losses for Real-Time Style Transfer and Super-Resolution. *arXiv* **2016**, arXiv:1603.08155.