Received: 5 April 2023 | Revised: 7 September 2023 | Accepted: 3 October 2023

DOI: 10.1002/qj.4592

Quarterly Journal of the Royal Meteorological Society RMetS

RESEARCH ARTICLE

Identifying and characterising trapped lee waves using deep learning techniques

Jonathan Coney[1] | Leif Denby[1,2] | Andrew N. Ross[1] | He Wang[3] | Simon Vosper[4] | Annelize van Niekerk[5] | Tom Dunstan[4] | Neil Hindley[6]

[1]Institute for Climate and Atmospheric Science, School of Earth and Environment, University of Leeds, Leeds, UK

[2]Weather Research Division, Danish Meteorological Institute, Copenhagen, Denmark

[3]Department of Computer Science, University College London, London, UK

[4]Met Office, Exeter, UK

[5]Research Department, European Centre for Medium Range Weather Forecasts, Reading, UK

[6]Centre for Climate Adaptation and Environment Research, University of Bath, Bath, UK

Correspondence
Jonathan Coney, School of Earth and Environment, University of Leeds, Leeds LS2 9JT, UK.
Email: mm16jdc@leeds.ac.uk

Funding information
Met Office; Natural Environment Research Council, Grant/Award Number: NE/S007458/1

Trapped lee waves, and resultant turbulent rotors downstream, present a hazard for aviation and land-based transport. Though high-resolution numerical weather prediction models can represent such phenomena, there is currently no simple and reliable automated method for detecting the extent and characteristics of these waves in model output. Spectral transform methods have traditionally been used to detect and characterise regions of wave activity in model and observational data; however, these methods can be slow and have their limitations. Machine-learning (ML) techniques offer a new and potentially fruitful method of tackling this problem. We demonstrate that a deep-learning model can be trained to accurately recognise and label coherent regions of lee waves from vertical velocity data on a single level from a high-resolution numerical weather prediction (NWP) model. Using transfer learning, wave characteristics (wavelength, orientation, and amplitude) can be extracted from the trained segmentation model. The use of synthetic wave fields with prescribed wave characteristics makes this transfer learning possible without the need to characterise real complex wave fields. Addition of noise to the synthetic data makes the models more robust when applied to more complex and noisy NWP data. The collection of trained models produced provides a valuable tool to investigate the prevalence and nature of lee wave activity, as well as a new way for forecasters to detect resolved waves. The deep-learning model was more capable and quicker at detecting and characterising lee waves than a spectral technique was. This work is just one example of how already established ML techniques can be used to detect and characterise complex weather phenomena from NWP model output and observational data, and how the careful use of synthetic data can reduce the requirements for large volumes of hand-labelled training data for ML models.

KEYWORDS

deep learning, mountain waves, trapped lee waves

1 INTRODUCTION

Deep learning, a type of machine learning (ML), is the discipline of training deep neural networks to autonomously extract nonlinear relationships between large quantities of data to produce a given prediction. One such task that deep learning can be applied to is segmentation; that is, an image classification problem where each pixel within an image is classified as belonging to a specific predefined class. An example of a deep- learning architecture for such segmentation is a U-Net, a type of convolutional neural network first described for the segmentation of medical imagery, for cell detection, and shape measurement (Ronneberger *et al.*, 2015). U-Nets are used for pixelwise predictions that can take the form of segmentation masks (Boolean 0/1) and classification (discrete) and regression (continuous) problems.

Since its introduction, the U-Net architecture has been used for image segmentation problems in a wide range of fields, including within Earth sciences. Examples include a land-cover classification from high-resolution satellite imagery over Beijing, identifying regions of buildings, water, roads, vegetation, and a separate classification for shadows, and "other" (Zhang *et al.*, 2018), recognition of regions of clouds within photographs of the sky (Dev *et al.*, 2019), and estimation of gravity-wave momentum fluxes at 100 hPa from low-resolution winds, temperature, and specific humidity at lower levels in the atmosphere using a 29-year reanalysis data set Matsuoka *et al.* (2020).

By using deep-learning methods to automate tasks traditionally only capable of being undertaken by humans or complex algorithms, ML models can be used on unseen data to analyse and identify features, such as hazardous weather phenomena. In addition, the application of deep-learning models to a large dataset—such as an archive of numerical weather prediction (NWP) model output—provides the opportunity to improve the understanding of weather phenomena by being able to analyse large datasets for cases of interest, a task that would be unfeasible by hand. Meteorological data, of which there are often large volumes on long time-scales, offer a prime data source for typically data-hungry ML techniques.

Trapped lee waves are one potential application where features are often difficult to identify by traditional means and where deep-learning methods offer the potential for new insights. Mountain waves (orographically generated internal gravity waves) are caused by the forced ascent of stably stratified air over orography (Durran, 2003). Depending on the vertical atmospheric profile, the waves may become vertically trapped within a layer of the atmosphere and can propagate horizontally in excess of 100 km beyond the mountain with horizontal wavelengths typically between 5 and 35 km (American Meteorological Society, 2012; Ralph *et al.*, 1997). These types of mountain waves are known as trapped lee waves (or lee waves for short).

Lee waves with a sufficiently large amplitude can induce strong downslope winds from topography and flow separation, where air close to the surface is transported aloft (Vosper *et al.*, 2006). The flow separation causes low-level turbulent vortices near the surface. These regions of strong turbulence, where the flow at low levels may be reversed compared with the background flow, are called rotors (Doyle and Durran, 2002). The turbulence and large wind shears associated with rotors on the lee side of a mountain are particularly hazardous for aircraft, especially during take-off and landing (Darby and Poulos, 2006). Gusts associated with waves can be problematic for road transport, and Vosper *et al.* (2013) noted that a number of wind-related incidents to high-sided vehicles have occurred in the lee side of the Pennine Hills in northern England. The turbulence that caused the crash of a commercial aeroplane over Mount Fuji in 1966 was attributed to mountain waves (International Civil Aviation Organization, 1968). Pilots need to be aware of lee waves and rotors: Ágústsson and Ólafsson (2014) discussed the severe turbulence experienced by an aircraft near Iceland in 2008, and the occurrence of strong turbulence hazardous to air traffic at Mount Pleasant Airfield on the Falkland Islands motivated the rotor study by Mobbs *et al.* (2005). Hence, forecasting centres are interested in identifying and communicating these risks accurately.

Lee waves can sometimes be observed during daylight hours in visual satellite imagery, with characteristic striped cloud patterns, caused by condensation of water vapour in air that has risen and cooled in the peak of the wave. When clouds are not present, or at night, visual satellite imagery cannot be used to observe lee waves. In place of using satellite imagery to detect the presence of gravity waves, these waves can be identified in output data from NWP models run at high enough spatial resolution to resolve the waves. For example, the Met Office's UKV model configuration (a high-resolution deterministic NWP model over the UK, which has a horizontal resolution of ~1.5 km in mid-latitudes; Tang *et al.*, 2013) has resolved lee waves since the dynamical core was upgraded in 2015, with improved numerical stability allowing the use of "reduced off-centring in the temporal discretization" (Elvidge *et al.*, 2017). Hence, the model more readily supports short wavelength gravity waves, without the need for "computationally expensive dynamics settings" (Sheridan *et al.*, 2017).

Lee waves are visible in the model output from a range of fields; however, vertical velocity just above the height of the topography is particularly helpful, as the background values of vertical velocity of atmospheric motion

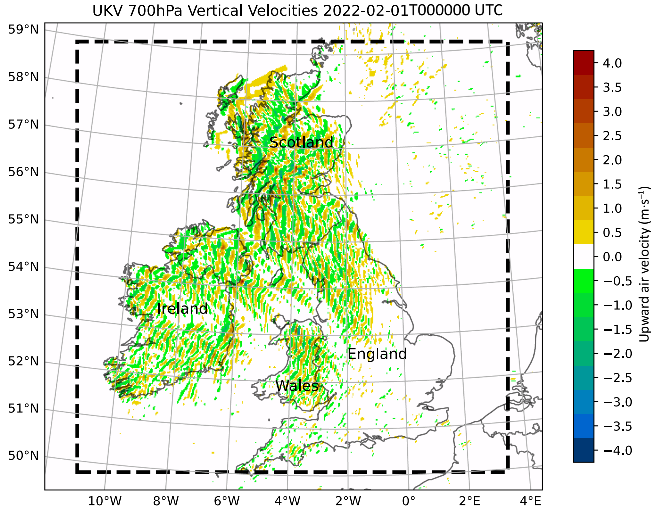

FIGURE 1 Example of the UKV 700 hPa vertical velocity analysis data, at 0000 UTC on February 1, 2022. The dashed box shows the region used for training data. Several lee wave patterns over Ireland, Wales, northern England, and Scotland can be discerned by eye. These regions are labelled to aid the reader unfamiliar with the geography. [Colour figure can be viewed at wileyonlinelibrary.com]

not associated with gravity waves are typically small and so the wave signal is clearer. Figure 1 shows an example of the UKV vertical velocity model output at 700 hPa over the UK where lee waves have been resolved. The striped vertical velocity pattern associated with waves, and their dominant orientation and wavelength, can be seen by eye, but there is also no operational method to automatically retrieve these characteristics from the NWP output. Typically, spectral techniques are used to detect and measure wave characteristics, such as the Fourier transform, wavelet transform (e.g., Hindley *et al.*, 2015), or the S-transform (e.g., Hindley *et al.*, 2019; Stockwell *et al.*, 1996). However, these methods assume an idealised mathematical representation of waves (such as planar, monochromatic waves) and cannot acquire a knowledge of "real-world" wave characteristics, where the physical scales, orientations, and frequencies of waves may vary within one wave cycle. An ML method to recognise and characterise the patterns of waves could provide a powerful and computationally inexpensive tool to post-process NWP output and remote-sensing observations for wave detection and characterisation. Knowledge of the characteristics of lee waves, in forecasts and from a climatological perspective, can help aid understanding of the relationship between waves, the orography, and the meteorology, as well as make it possible to build automatic early-warning systems for rotor activity.

This work demonstrates the application of a U-Net deep-learning model to recognise and segment regions of lee wave activity from high-resolution NWP model output. The trained model is then retrained using synthetic data to diagnose characteristics about the real waves seen in the UKV data. An overview of the methodology is given in Section 2; the results are presented in Section 3; and conclusions are given in Section 4.

2 | METHODOLOGY

Previous work by Sheridan *et al.* (2017) has shown that the characteristics and impacts of lee waves in the operational Met Office UKV model is in agreement with observations. Using NWP model data provides spatially dense and continuous coverage over a long time period, regardless of meteorological or daylight conditions, unlike the use of satellite imagery to observe lee waves from wave clouds. Visible or infrared satellite imagery is an unreliable indicator of lee waves because it requires cloud to be coincident with lee wave motion; and even then, higher level cloud may mask the lee wave cloud patterns. By using NWP model output, most lee wave cases can be identified without biasing the sample to conditions with suitable cloud cover.

The approach used here to detect lee waves using a neural network relied on having some explicit "truth" data as a target. This is known as supervised learning. In this case, the truth data were a hand-labelled mask of each vertical velocity snapshot containing the location of wave packets that the ML model tries to predict. Though human labellers are good at differentiating between waves and other sources of vertical motion (e.g., convection), precisely identifying where the boundary of the wave packets is is difficult by eye when labelling the data. For example, in Figure 1, it is not a trivial task to decide for every pixel, especially those on the edge of a wave region, which pixels contain a wave and which do not. Therefore, it is important to assess whether the trained neural network has learned to recognise what lee waves look like rather than learning the precise boundaries of wave regions in the hand-labelled masks.

2.1 | Data

The Met Office have archived the output of their operational UKV model since 2018, and a subset of this archive of hourly data was used to train and test a model to learn to identify and characterise the patterns of lee waves. Vertical velocity analysis data over Britain and Ireland on the 700 hPa surface from the Met Office UKV output were obtained from January 1, 2018, until June 30, 2022, and data from January, February, and July 2021 were labelled by hand. The 700 hPa data have been used before for lee

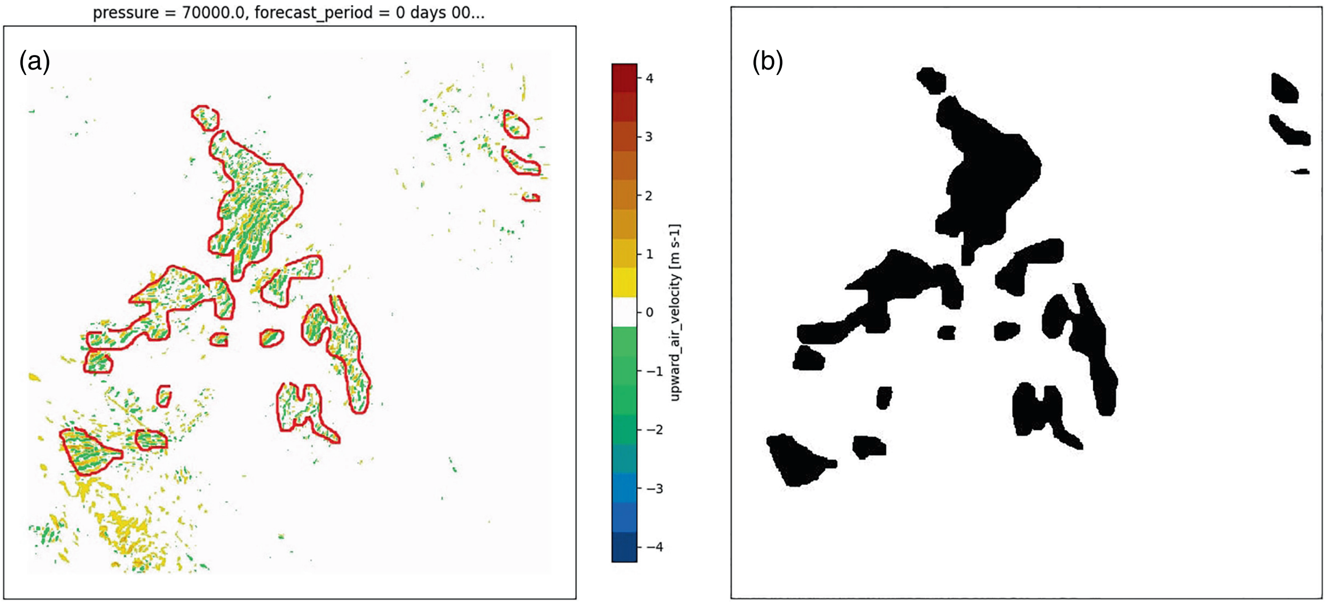
pressure = 70000.0, forecast_period = 0 days 00...
(a)
4
3
2
1
0
−1
−2
−3
−4
upward_air_velocity [m s-1]
(b)

FIGURE 2 Overview of labelling process for an example of test-set data. (a) Vertical velocity data without coastline overlaid, and hand-drawn regions of wave activity outlined. (b) Produced segmentation mask: white (0), no waves; black (1), waves. [Colour figure can be viewed at wileyonlinelibrary.com]

wave detection from model data over the UK, such as by Vosper *et al.* (2013). The 700 hPa surface is above the height of the orography in the UK, and so incorrectly interpolated values of vertical velocity within the data due to the chosen pressure surface intersecting with the orography are avoided.

The model data were regridded from their native variable-resolution grid—1.5 km on the inner domain and 4 km on the boundaries; details in Tang *et al.* (2013)—to a 2 km regular grid prior to archival. Although some detail may have been lost in the regridding from 1.5 km to 2 km, characteristics such as wavelength were more easily diagnosed on a regular fixed-resolution grid than on a variable-resolution grid. The concept of effective resolution suggests that at least six grid points are needed in order to represent a wave, and so waves with wavelength over 12 km should be detectable (Sheridan *et al.*, 2017).

The data were split into sets, one used for training and the other to test the skill of the trained model after training. The training and test vertical velocity slices were labelled to produce binary segmentation masks with 0s for pixels with no waves, and 1s for pixels where there was a wave. The wider forecast area was cropped to 512 × 512 px^2 (1,024 × 1,024 km^2) to create square training data, but the cropped area still contained the entirety of Britain and Ireland (shown by the dashed box in Figure 1). Binary segmentation masks of 512 × 512 px^2 were created using a custom Jupyter notebook utilising Matplotlib interactive notebook functionality. A 512 × 512 px^2 array containing 0 everywhere was created. Then, the human labeller drew around the regions they wished to label as a wave (regions in red in Figure 2a). Pixels within each hand-drawn shape (closed by drawing a line from the last point to the first) were then changed to 1 from 0 in the mask array. When done, the binary mask containing 0s for regions with no waves and 1s for regions containing waves was saved (Figure 2b).

Training data comprised pairs of 335 vertical velocity cross-sections at 700 hPa from January 1–18, 2021, covering different times of day, and corresponding binary segmentation masks. Some examples containing no wave activity were excluded from the training set so that the number of samples with and without waves was similar, in order not to encourage the model to never predict waves. Despite these measures, there was still a class imbalance in the training set where 10% of pixels were labelled as waves. Two test datasets were created, one from February 2021 and one from July 2021. These test sets contained vertical velocity cross-sections and segmentation masks from 0900 UTC each day within the respective month. The purpose of the July set was to check that the trained model had not learned to identify waves only within winter months. Some 16% of pixels were labelled as waves in the February test set, and 3% of pixels were labelled as waves in the July test set. Although there could be some variation between months on the range and scale of waves, this should be unimportant because the model was being trained to recognise the pattern of waves.

For wave characteristic prediction, creating a dataset from model output with pixelwise true wave

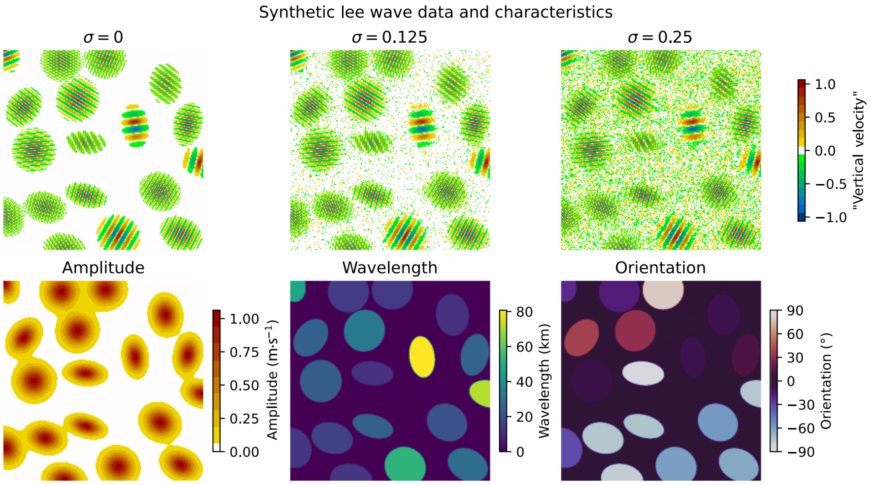

FIGURE 3 Example synthetic normalised vertical velocity input data generated for lee wave characteristic learning, at different standard deviations of normally distributed noise and output wave characteristics that the model was tasked to predict. For wavelength and orientation learning, the amplitude was set to 1. For amplitude learning, this value varied between 1 and 5. [Colour figure can be viewed at wileyonlinelibrary.com]

characteristics would be exceedingly difficult, and perhaps impossible to do correctly. This is often a challenge in supervised learning applications. To work around this, synthetic data were created with explicitly known wavelength, orientation, and amplitude, with characteristics selected to mimic the gravity waves seen by eye in the UKV data.

The synthetic data were generated by placing non-intersecting ellipses of differing sizes at random locations within an image of $512 \times 512\,\text{px}^2$ (Denby, 2023). Each ellipse contained a Gaussian wave packet of regular cosine waves with an orientation chosen from a uniform distribution between 0° and 180° to ensure all orientations were covered equally. Wavelength was chosen at random from a chi-squared distribution with two degrees of freedom. Wavelengths up to 80 km were used, so the synthetic data more than spanned typical lee wave wavelengths of 5–35 km (American Meteorological Society, 2012), with the chi-squared distribution ensuring that there were more examples of waves in the typical range of wavelengths than there were exceeding 35 km, and that all wavelengths were positive. The amplitude decays to the edge of the wave packets to simulate waves decaying, as seen in the vertical velocity NWP model data. For each example of synthetic data, a number of wave packets with different orientations, amplitudes, and wavelengths were produced. For wavelength and orientation prediction, amplitudes were kept constant in all wave packets. For amplitude prediction, synthetic wave amplitude varied in the range 1–5 $\text{m}\cdot\text{s}^{-1}$, consistent with waves observed in the UKV data.

Pixelwise random Gaussian noise from the normal distribution with mean zero and standard deviation σ was also added to the data in order to train models that were more robust to realistic gravity waves embedded alongside other sources of vertical atmospheric motion. σ took values between 0 and 1. The noise array was then added to the synthetic data to produce noisy data. Several examples of the noisy data for $\sigma = 0.25, 0.5, 1$, and an example without noise ($\sigma = 0$) are shown in Figure 3. This ensured a range of noisy data, from $\sigma = 0$ with no noise through to $\sigma = 1$ where the amplitude of the waves is the same as the amplitude of the noise.

It is highly likely that, in reality, the non-wave sources of vertical motion ("noise") in the UKV data are correlated to the waves. More complicated methods of approximating this distribution in the UKV data may produce better synthetic data, and better models; but, for simplicity, Gaussian noise was used here.

2.2 | Network architecture and model training

The basis of the deep-learning models used here is the U-Net, a type of neural network commonly used for segmentation problems, which takes two-dimensional data and makes pixelwise predictions. A U-Net consists of two main parts, an encoder and a decoder. The encoder (or backbone) can extract spatially complex patterns by coarse-graining the input data with increasing depth in the network and compositing learned features. The number of channels (the scalar values at a single spatial location) is increased through the encoder to maintain information capacity, to account for the data being spatially coarse-grained in the encoder. The decoder uses the patterns extracted by the encoder and upsamples the data while reducing the number of channels (decreasing the depth), so that a prediction can be made for each pixel. In addition, the upsampled data are combined with data from the encoder at the same level (skip-connections). These skip-connections allow the U-Net to retain high spatial fidelity by combining the up-scaled values in the decoder with more spatially dense values from the encoder. Finally,

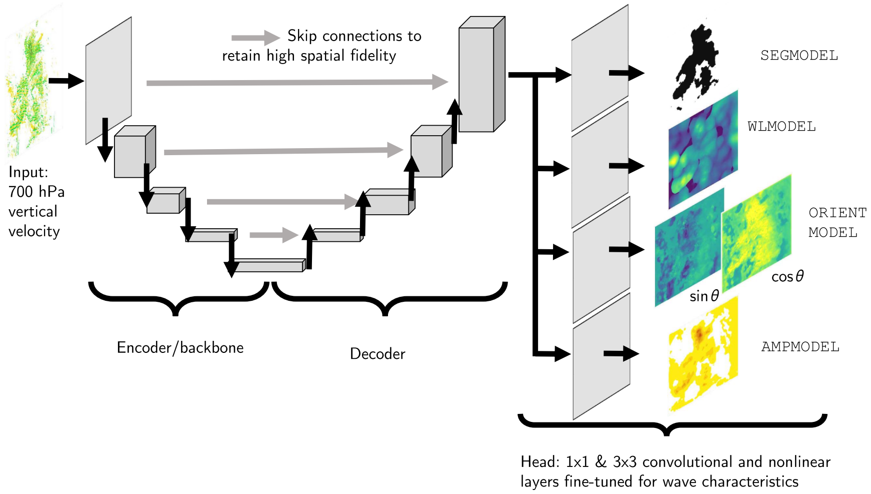
Skip connections to retain high spatial fidelity
Input: 700 hPa vertical velocity
Encoder/backbone
Decoder
SEGMODEL
WLMODEL
ORIENT MODEL
sin θ
cos θ
AMPMODEL
Head: 1x1 & 3x3 convolutional and nonlinear layers fine-tuned for wave characteristics

FIGURE 4 Overview of the U-Nets used. The encoder and decoder parts are shown, as well as the head of the model for making final predictions based on the features extracted. The full model is available in the Supporting Information. [Colour figure can be viewed at wileyonlinelibrary.com]

the head is where the remaining pixelwise learnt spatial features are further manipulated through pixelwise transforms to produce predictions per pixel.

The variation of the U-Net used here was the "Dynamic U-Net" implemented in the Python library fastai (Howard and Gugger, 2020). This uses a model designed to extract patterns from data (a Resnet34, pretrained on the ImageNet dataset; Deng *et al.*, 2009; He *et al.*, 2016) as an encoder, which means that the part of the model dedicated to extracting patterns from data is already trained to do this. Fastai is a wrapper for the Python deep-learning library PyTorch (Paszke *et al.*, 2019). Fastai allows an approach to deep learning that is understandable and easy to access for a user with limited experience with deep learning, yet it still produces accurate deep-learning models. A simplified overview of the U-Net used in this study is shown in Figure 4.

2.3 | Segmentation

The segmentation model (referred to as SEGMODEL from this point) takes the two-dimensional regridded 700 hPa vertical velocity field as an input and outputs a two-dimensional Boolean mask of the same shape, containing a prediction of where gravity waves are present. Supervised learning was used to train SEGMODEL, and so labelled data were needed, as described in Section 2.1. The labelled training set was divided randomly into a train set (80% of the data) and a validation set (20% of the data), during training. This prevented overtraining by stopping training while the trained model performed similarly on the train and validation set. The test set was only used to assess how well the trained SEGMODEL performed at segmenting lee waves on data not used during training.

The training data were augmented using the built-in fastai augmentation functions, including flipping, rotation (up to 360° and a probability of 0.9 of any rotation being applied), and zooming (up to 20× and probability of 0.5 of any zoom being applied) of the data. The vertical velocity data were normalised to have a mean of 0 and standard deviation of 1. Augmentation of the data minimises overfitting of the model during training (Shorten and Khoshgoftaar, 2019). By augmenting the data, the model was exposed to waves at a range of wavelengths and orientations during training, beyond the original training data. For example, by rotating the lee waves data, the model can learn waves at a variety of angles, not just the waves in the typical southwesterly flow over the UK. Zooming should allow the model to learn to recognise waves of longer wavelengths than those available in the training data. Waves are generated through the same mechanism regardless of the orientation, so rotating the data during training should not affect learning negatively.

Cross-entropy loss, which prefers models with a high degree of confidence in their predictions, was used as the loss function as it is well suited for Boolean prediction problems such as this (Jadon, 2020), rather than other metrics such as the Jaccard score, used later for model evaluation. To prevent overtraining, training continued until the validation loss appeared to be increasing again, using the built-in fastai early stopping callback, with an epoch window (patience) of five epochs. Once the SEGMODEL was trained, its performance was assessed using the unseen test dataset, consisting of 28 examples of vertical velocity data and hand-labelled lee waves from February 2021 and 31 examples from July 2021.

2.4 | Wave characteristics

Given that the neural network has learned to recognise lee waves during training, then it should also have learned something about the wave characteristics. By fine-tuning only the layers in the "head" on synthetic wave data with known characteristics, a model can be produced that predicts a wave characteristic instead of a segmentation. This is an efficient way of training networks to extract multiple characteristics and, if successful, supports the hypothesis that the original model was learning some properties of the waves.

Three copies of the trained SEGMODEL were taken: WLMODEL to predict wavelength; ORIENTMODEL to predict wave orientation; and AMPMODEL to predict wave amplitude. In each of the copies, the weights in the encoder and decoder were frozen. Only the weights of the layers in the "head" of the copies (labelled as such in Figure 4), consisting of nonlinear scaling functions, 1×1 and 3×3 convolutional layers, were trained (fine-tuned) on the synthetic data (see Section 2.1) to predict the desired characteristic. In general, these layers in the "head" transform U-Net feature vectors into predictions. By freezing the weights of the spatial-feature-extracting "encoder" in the model, the contextual information about waves in UKV data that the SEGMODEL had learned was retained.

2.4.1 | Wavelength and orientation

The wavelength network WLMODEL was trained to predict the wavelength in kilometres, whereas the orientation network ORIENTMODEL predicted the sine and cosine of a wave's orientation so that the orientation could be recovered using the arctangent. By predicting the sine and cosine of the orientation (the direction of wave propagation, perpendicular to the wave-fronts) rather than the wave direction in degrees, the discontinuity of angles around 0° and 360° was avoided. Orientation was predicted in the range $(-90°, +90°]$ because waves with orientation 180° apart look identical. Lee waves are often quasi-stationary, and even for propagating waves the direction and speed cannot be calculated with single time snapshots of data. Since the wave characteristics were not available in the UKV training data, synthetic data were created to retrain these models, as outlined in Section 2.1. We created 550 examples of data, with explicitly known orientation and wavelength. We used 500 examples as training data (with a 80%/20% train/validation split) and an additional 50 examples were used as test data. In a similar vein to training the SEGMODEL, the altered models WLMODEL and ORIENTMODEL were trained using the early stopping callback, with an epoch window (patience) of 5. The loss function used was the mean-squared error, as these tasks are regression rather than classification.

2.4.2 | The S-transform

The neural-network-predicted wave characteristics (wavelength, orientation, amplitude) were compared against characteristics derived using an established spectral analysis technique: the one-dimensional S-transform (Stockwell *et al.*, 1996). The S-transform provides time–frequency or distance–wavelength localisation of signals present in input data and is thus ideally suited for the measurement of gravity wave packets. The S-transform therefore provides an existing spectral technique with which to compare the predictive skill of the neural networks, but note that the S-transform measurements should not necessarily be taken as "truth". The S-transform is being used here as a means of verifying that the ML models are producing wave characteristics that are realistic and reasonable in line with an existing technique for deriving characteristics of gravity waves.

Here, the two-dimensional S-transform application developed by Hindley *et al.* (2016) and Hindley *et al.* (2019) is used, which provides the dominant local spectral properties (wave amplitude, wavelength, orientation) at every pixel of the input image, but then restricted to those regions recognised as waves by the SEGMODEL. This gives undue credit to the S-transform, which does not segment waves into wave and non-wave regions as the SEGMODEL does.

One feature of the S-transform application of Hindley *et al.* (2019) used here is that it can be tuned to provide improved performance for waves present in a given dataset. Specifically, the analysis first computes the fast (discrete) Fourier transform and selects the N elements (number of frequency voices) with the largest spectral power for further localisation analysis. The larger the number of frequencies N, the higher the fidelity of the analysis but the longer the run time. For images containing simplified large-scale monochromatic waves, only small values of N are required, but higher N values can be useful for images with numerous small-scale waves with complex structures. Second, a scaling parameter c can be used to tune the spectral sensitivity of the S-transform. From a default of $c = 1$, increasing c improves spectral localisation at the expense of spatial localisation, whereas decreasing c achieves the opposite.

The S-transform was applied to the synthetic and NWP vertical velocity data with three different numbers of frequency voices ($N = 15, 80{,}150$) to determine the most appropriate value to capture all relevant waves in the data. A scaling parameter of $c = 0.25$ was used for this initial test, as used in previous studies (Hindley *et al.*, 2020; Wright *et al.*, 2017). Later, the value of c was adjusted to calculate an optimal value to use for further analysis. Using $N = 15$ frequency voices resulted in a cut-off in the output

TABLE 1 Comparison of R^2 least-squares correlation coefficient for wavelength and orientation derivation using S-transforms with scaling parameter $c = 0.25$, three values for the number of frequency voices N used in the spectral analysis, and machine-learned WLMODEL and ORIENTMODEL versus known truth on synthetic data without noise.

	Least squares linear regression coefficient R^2 (three decimal places)			
Characteristics	**Wavelength**	**Orientation: sin**	**Orientation: cos**	**Amplitude**
WLMODEL	0.997			
ORIENTMODEL		0.961	0.979	
AMPMODEL				0.997
S-transform ($N = 15$)	0.042	0.198	0.071	
S-transform ($N = 80$)	0.969	0.974	0.941	
S-transform ($N = 150$)	0.968	0.974	0.940	

wavelength, with shortest wavelengths of 50 km, whereas frequency voices of $N = 80$ and above resulted in a cut-off around 10 km. Given that the horizontal spacing of the vertical velocity data is 2 km, a cut-off of 10 km is an appropriate limit for the smallest wavelength that can be reliably measured.

To measure the ability of the ML models and S-transform to reconstruct the waves, the least-squares correlation coefficient R^2 with the ML wavelengths on the synthetic data was computed. The R^2 values for the three sets of S-transform characteristics and the machine-learned characteristics were compared against the true values and are given in Table 1. From these tests a value of $N = 80$ or 150 frequency voices gave a similarly good correlation against the true wavelength and orientation, but 80 frequency voices was computationally cheaper and so is used in the remainder of the article.

2.4.3 | Wavelength and orientation model selection

The synthetic data were much simpler than the original UKV data they were attempting to replicate, as the synthetic data did not contain other sources of vertical velocity, nor superposition of waves. Since the SEGMODEL could detect where gravity waves are, it must be able to extract the salient features of gravity waves, irrespective of whether there are other physical processes creating vertical velocity variations. Originally, the synthetic data were noiseless, as it was thought that the SEGMODEL had retained sufficient learning from its initial training to handle the vertical velocities in the UKV data not associated with lee waves when predicting characteristics. However, the wavelengths predicted by the model trained on noiseless data produced unrealistically long wavelengths on the UKV data. To address this, seven more wavelength models were trained on noisy data at a range of standard deviations σ in [0.125, 0.25, 0.375, 0.5, 0.6, 0.8, 1]. Some examples of the noisy data are shown in Figure 3.

The performance of all the ML models and the S-transforms were compared for different noise levels. Figure 5shows the performance of the eight ML wavelength models and three S-transforms with 80 frequency voices and $c = 0.25$, $c = 1$, and $c = 4$. The best-performing ML model in each case was the model that was trained on the data most similar in noise level to the corresponding test set at lower values of σ. At values of $\sigma > 0.5$ the picture is less clear, with RMSEs higher than for values of $\sigma \leq 0.5$. Models trained with noise performed better on data without noise than the model trained with no noise performed on data with noise, suggesting that adding noise during training allows the model to better generalise to different levels of noise. Over a range of noise amplitudes ($\sigma = 0.125$–0.5), the models are fairly robust at accurately predicting the wavelength from data with different noise levels within this range, as shown by the plots in Figure 5. The model trained on noise $\sigma = 0.125$ has $R^2 > 0.8$ for all apart from the noisiest synthetic data. This suggests that the WLMODEL trained on no noise was overfitted to the training data (expecting a specific level of noise) and rapidly decreases in skill compared with models trained with some noise. The addition of noise to the training data seems to have mitigated this overfitting. Though the Gaussian noise in the synthetic data is relatively simplistic compared with the correlated non-wave sources of vertical velocity in the UKV data, the trained models were all exposed to the non-wave sources of vertical velocity during the training of the SEGMODEL.

At $\sigma < 0.5$ the S-transform wavelengths showed good correlation with the true wavelengths, with $R^2 > 0.8$. However, the S-transform results consistently had a worse least-squares correlation coefficient than the best-performing ML model, which is likely due to the limited frequency voices (and thus orientations) of using the discrete Fourier transform to calculate the S-transform

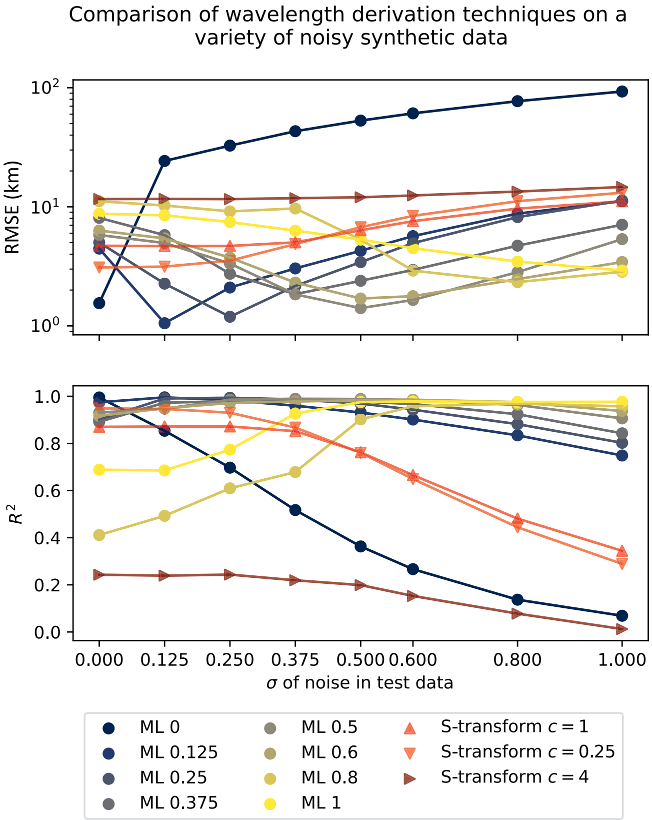

FIGURE 5 Comparison of the root-mean-squared error (RMSE; smaller values better) and least-squares regression coefficient R^2 (larger values better) performance of wavelength derivation techniques compared with the truth (three S-transforms with 80 frequency voices and different scaling parameters c; and eight machine-learning (ML) models) on synthetic data with different levels of noise. Then, all are applied to synthetic data with those levels of noise. The number after "ML" indicates the σ of the noise the model was trained on. [Colour figure can be viewed at wileyonlinelibrary.com]

(Hindley *et al.*, 2019), which results in a slightly less "exact fit" for the input waves.

Figure 6 shows a comparison of the orientation derivation techniques on the synthetic data. Owing to the circular nature of the data, the two methods used in Figure 6 to analyse the accuracy of the predictions are the Euclidean distance and the least-squares regression coefficient R^2 for the cosine of the orientation. The Euclidean distance metric E between the angles θ and ϕ is defined as

$$E(\theta,\phi) = \sqrt{(\cos\theta - \cos\phi)^2 + (\sin\theta - \sin\phi)^2}.$$

For example, a pair of angles 90° out of phase with each other would have $E = \sqrt{2}$, and a pair of angles 30° apart would have $E = 2 - \sqrt{3} \approx 0.27$. The R^2 plotted is of the cosine of the angles to avoid discontinuity around −90° and +90°.

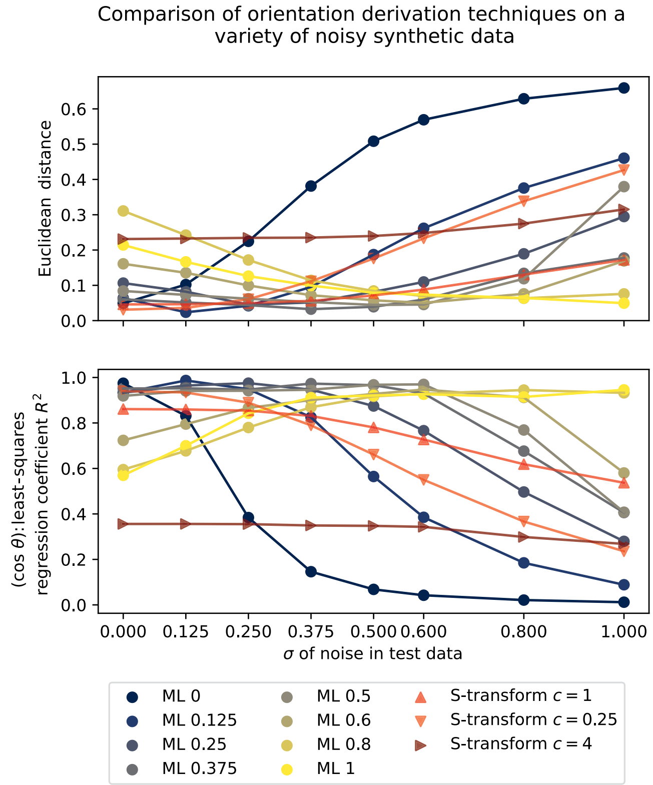

FIGURE 6 Comparison of the Euclidean distance (smaller values better) and least-squares regression coefficient R^2 (larger values better) performance of orientation derivation techniques compared with the truth (both machine-learning [ML] and S-transforms as in Figure 5). [Colour figure can be viewed at wileyonlinelibrary.com]

There is a smaller difference between the ML and S-transform techniques in Figure 6 than in Figure 5, with the S-transform $c = 1$ performing better than some of the ML models in the synthetic test data. The skill of an ML model for wave orientation decreases more rapidly when applied to data with other noise levels compared with the wavelength models. That said, the orientation models trained on any noise are more robust to other noise levels than the orientation model trained on noiseless data. The model trained on noisy data with $\sigma = 0.25$ has a least-squares coefficient > 0.8 for data up until $\sigma = 0.6$, so is robust to a range of noise amplitudes.

From these tests, the WLMODEL trained on data with noise $\sigma = 0.125$ was used to predict UKV wavelengths, and the ORIENTMODEL trained on data with noise $\sigma = 0.25$ was used to predict UKV orientation. To select which c to use in the S-transform, local variations in wavelength with $c = 0.25$ were found to be too large, whereas choosing $c = 1$ resulted in a spatially smoother wavelength field. Setting $c = 4$ resulted in oversmoothing that produced inaccurate wavelength estimates. Hence, $c = 1$ was used for future comparisons in synthetic and model data.

TABLE 2 Comparison between the U-Net and S-transform, and the time taken for the two methods to produce wave characteristics, for each example in the UKV February 2021 test set.

Method	Mean time taken (s)	σ (s)
U-Net	12.1	2.61
S-transform	150.9	12.3

2.4.4 | Amplitude

A neural network (fine-tuned in the same way as already described herein for wavelength and orientation) was used to extract the amplitude of the waves from the UKV data. The neural network was trained on synthetic data but with wave packets of variable amplitudes in between 1 and 5 m⋅s^{-1}. The synthetic data had a small amount of noise ($\sigma = 0.0625$) added, which resulted in a smoother amplitude model prediction over the UK, compared with a model trained on synthetic data without additional noise. The amplitude of observed waves and the synthetic data decays towards the edge of each wave packet, so the model was trained to predict this smooth envelope. On the synthetic test data, the trained amplitude model scored an R^2 of 0.999.

3 | RESULTS

This section presents the results of the segmentation model SEGMODEL against the hand-labelled truth, and the results of the wave characteristics models (WLMODEL, ORIENTMODEL, and AMPMODEL), with the wavelength, orientation, and amplitude output compared against those from the S-transform. The ML models ran significantly faster than the S-transform on the UKV test data. For example, it took an 1 hr 10 min for a standard laptop CPU to produce the S-transformed data for the 28 examples in the February UKV test set, whereas it took the same laptop 5.5 min to produce the wave mask, wavelength, orientation, and amplitude for the same set, a speed-up of 12.7×. Table 2 shows the mean and standard deviation of the time taken for the two methods to produce wave characteristics for each example of the February 2021 test data.

3.1 | Lee wave segmentation

Figure 7 shows two examples of lee wave segmentation on vertical velocity data: one example of test data from February 2021 (Figure 7a) and another from July 2021 (Figure 7b). These results show that the model has learned typical patterns of gravity waves during training. The SEGMODEL is skilful at recognising waves and is capable of ignoring non-wave sources of vertical velocity. This is evidenced in Figure 7a, which shows an occasion where there are large regions of waves in the data, which the SEGMODEL has recognised as waves. The area to the north of Ireland where the vertical velocity patterns look very different is likely to be convection and not wave activity, as there is precipitation in the model associated with it. The SEGMODEL correctly did not classify these regions as lee waves. Figure 7b shows an example where waves are apparent over Ireland, with smaller regions with wave-like features elsewhere.

Two test sets were used to analyse the performance of the trained SEGMODEL. One was from February 2021 and one was from July 2021. This was to check that the model was able to recognise waves from throughout the year. There are typically fewer waves in summer months, so the results from the two months are presented separately. As a reference, the output from a baseline “model” that always returned no waves everywhere (the ZEROS model) are presented alongside the results from SEGMODEL.

Four metrics of model performance on the test sets are summarised in Table 3: the pixel accuracy, Jaccard score, precision score, and recall score. The pixel accuracy is the percentage of pixels that were correctly identified by the model, compared with the hand-labelled truth. The Jaccard score (or intersection over union) is given by

$$\text{Jaccard score} = \frac{\left(\frac{|P_0 \cap T_0|}{|P_0 \cup T_0|} + \frac{|P_1 \cap T_1|}{|P_1 \cup T_1|}\right)}{2},$$

where P_i is the model's prediction and T_i is the hand-labelled truth for the ith class of pixel ($i = 0$: no wave; $i = 1$: wave). The Jaccard score is computed for each class by the area of overlap divided by the area of the union of the model's prediction and the truth. Then, the mean of these is taken to find the Jaccard score for the example of test data. The score shows how similar the prediction is to the hand labels, and therefore how good the model is. Although it is feasible to have used the Jaccard score as a loss function, the model was trained using cross-entropy owing to its good performance as a loss function in segmentation tasks (Jadon, 2020). The precision score is the number of true positives (correctly identified waves) divided by the number of true positives plus the number of false positives, and the recall score is the number of true positives divided by the number of true positives plus the number of false negatives (Pedregosa *et al.*, 2011).

Table 3 shows that on the February 2021 test set the trained SEGMODEL performed at 95% pixel accuracy. In this test set, only 16.9% of pixels were labelled as waves,

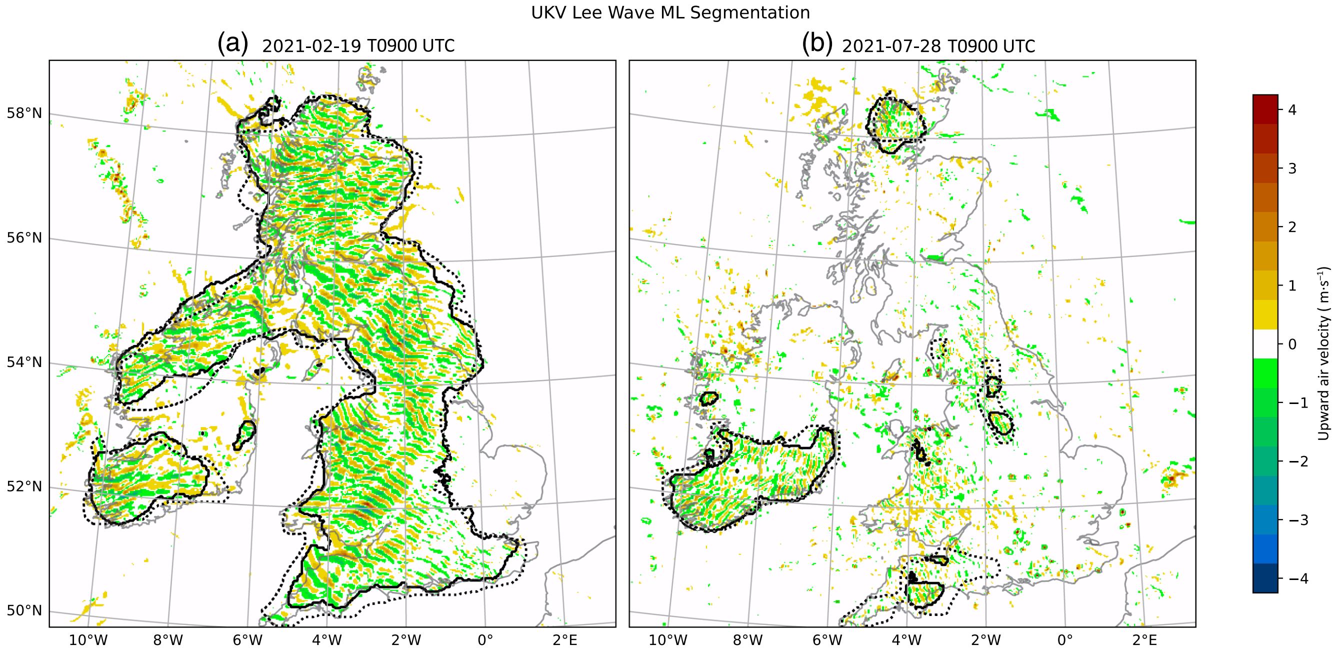

FIGURE 7 Two examples of the SEGMODEL-predicted lee wave segmentation over the UK. Vertical velocities at 700 hPa are shown in filled contours, with wave regions predicted by the model shown by the black line contour. The dotted line contour shows the hand-labelled waves. (a) An example from the test set in February 2021. Against the "truth" data, the segmentation achieved a pixel accuracy of 94% and a Jaccard score of 0.87. (b) An example of data from July 2021 containing waves segmented by the model with a pixel accuracy of 97% and a Jaccard score of 0.78. [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 3 Performance on the February 2021 and July 2021 test sets, by the SEGMODEL U-Net and ZEROS (a model that never predicts waves).

Model	Pixel accuracy (%)	Jaccard score	Precision	Recall
February 2021 test set				
SEGMODEL	95.0	0.78	0.87	0.70
ZEROS	84.4	0.42	0.0	0.0
July 2021 test set				
SEGMODEL	98.0	0.66	0.48	0.42
ZEROS	97	0.49	0.0	0.0

even though this was from a winter period with higher wave activity. The ZEROS model "performs" well in the pixel accuracy metric in the February test set, reflecting the small amount of wave activity compared with the background. The SEGMODEL has a Jaccard score of 0.78 compared with ZEROS with a Jaccard score of 0.42 in this set. The SEGMODEL is demonstrating skill by detecting waves in plausible locations compared with never predicting waves. Gravity waves have, by their very nature, a decaying amplitude envelope that makes defining a hard edge to a gravity wave envelope a poorly defined problem, which in turn means that exactly achieving a Jaccard score of 1 would be difficult. The precision and recall scores indicate a reasonable ratio between the number of correctly identified pixels and those incorrectly identified by the SEGMODEL. The ZEROS model scores 0 for both of these metrics since it did not correctly identify any waves.

On the July 2021 test set, Table 3 shows that waves occurred very infrequently from the pixel accuracy of the ZEROS model. However, the difference in Jaccard score shows that the SEGMODEL outperforms the ZEROS model by localising waves when they occurred. This infrequency of wave occurrence is reflected in the precision and recall scores.

3.2 | Lee wave characteristics

Having demonstrated the accuracy of the lee waves prediction network, the segmentation U-Net model is next utilised to infer lee wave characteristics: wavelength, orientation, and amplitude. These predicted characteristics were then restricted to regions containing waves using the predicted wave masks from earlier.

As discussed in Section 2.4, transfer learning was used to fine-tune the final layers of the SEGMODEL network to learn wavelengths (WLMODEL), orientation (ORIENTMODEL), and amplitude (AMPMODEL) of the waves. Figure 8 shows one such example of the predicted characteristics.

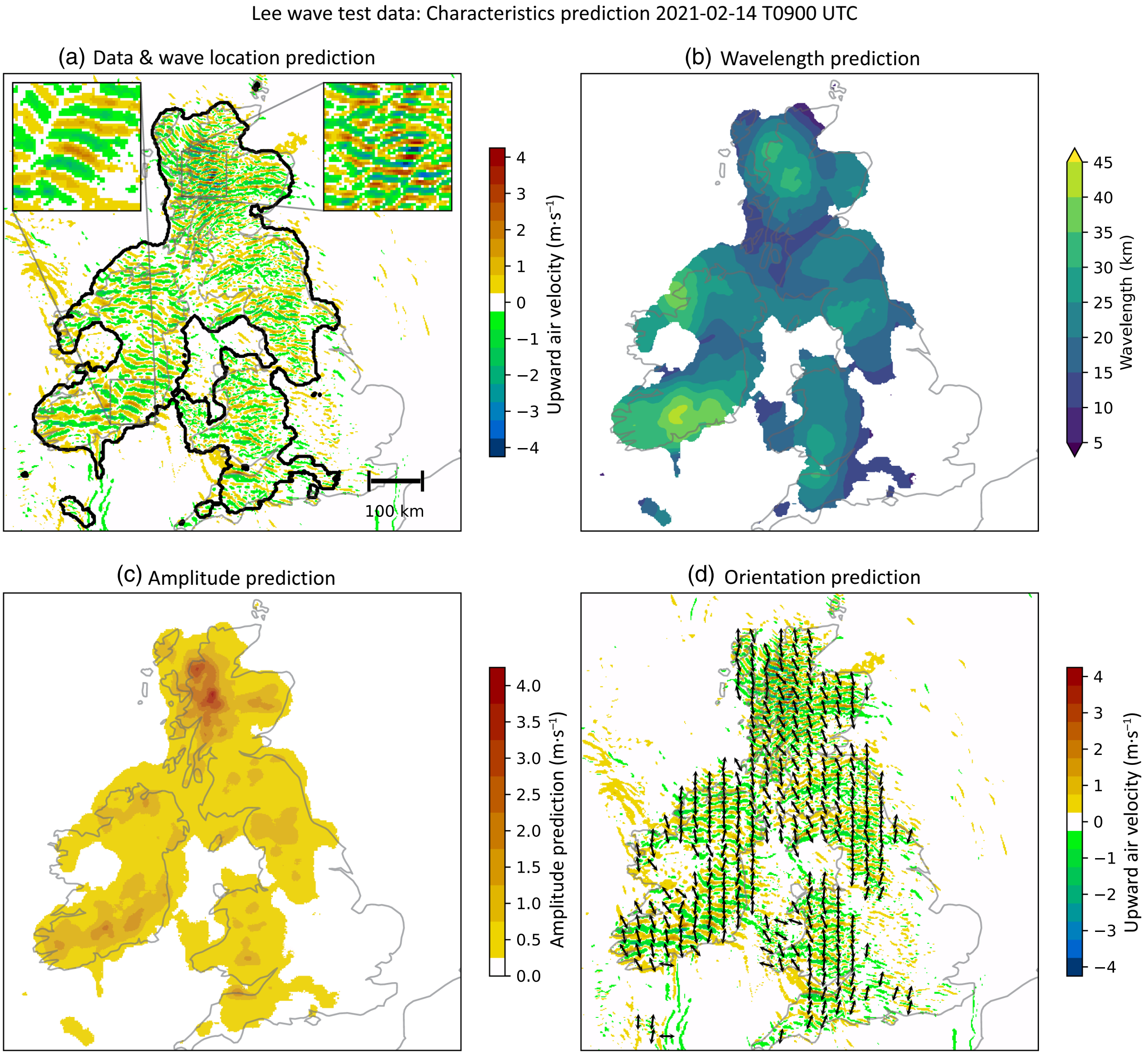
Lee wave test data: Characteristics prediction 2021-02-14 T0900 UTC
(a) Data & wave location prediction
100 km
Upward air velocity (m·s^{-1})
4
3
2
1
0
−1
−2
−3
−4
(b) Wavelength prediction
Wavelength (km)
45
40
35
30
25
20
15
10
5
(c) Amplitude prediction
Amplitude prediction (m·s^{-1})
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
(d) Orientation prediction
Upward air velocity (m·s^{-1})
4
3
2
1
0
−1
−2
−3
−4

FIGURE 8 Example of lee wave characteristic prediction with (a) vertical velocity and predicted lee wave regions (black line contour), (b) predicted wavelength for lee wave regions (model trained on data with noise $\sigma = 0.125$), (c) predicted wave amplitude for lee wave regions, and (d) orientation of lee waves for lee wave regions (perpendicular to wave-fronts) (model trained on data with noise $\sigma = 0.25$). The two inset regions in (a) demonstrate the difference in wavelength, amplitude, and orientation in Scotland and Ireland for this particular case. [Colour figure can be viewed at wileyonlinelibrary.com]

The predicted characteristics were compared against a spectral technique, the S-transform, in order to have a method to compare characteristic predictions against. The following subsections will deal with each characteristic in turn.

3.2.1 | Wavelength

The ML model and S-transform approaches were contrasted both on synthetic wavelength data samples (where the true wavelength value is known) and UKV simulation output (where the true value is not known). If the ML approach works well on the synthetic data compared with an S-transform, then this gives confidence that the ML-derived wavelengths from UKV data are reasonable. Though the S-transform-derived wavelengths cannot necessarily be regarded as "truth", they can be used to ensure that the ML model is consistent with the S-transform and produces physically realistic wavelengths.

Figure 9a,b shows a two-dimensional histogram for the synthetic test dataset, comparing the ML model predictions and S-transform ($N = 80$ frequency voices and $c = 1$) derived wavelengths against the true wavelengths. The

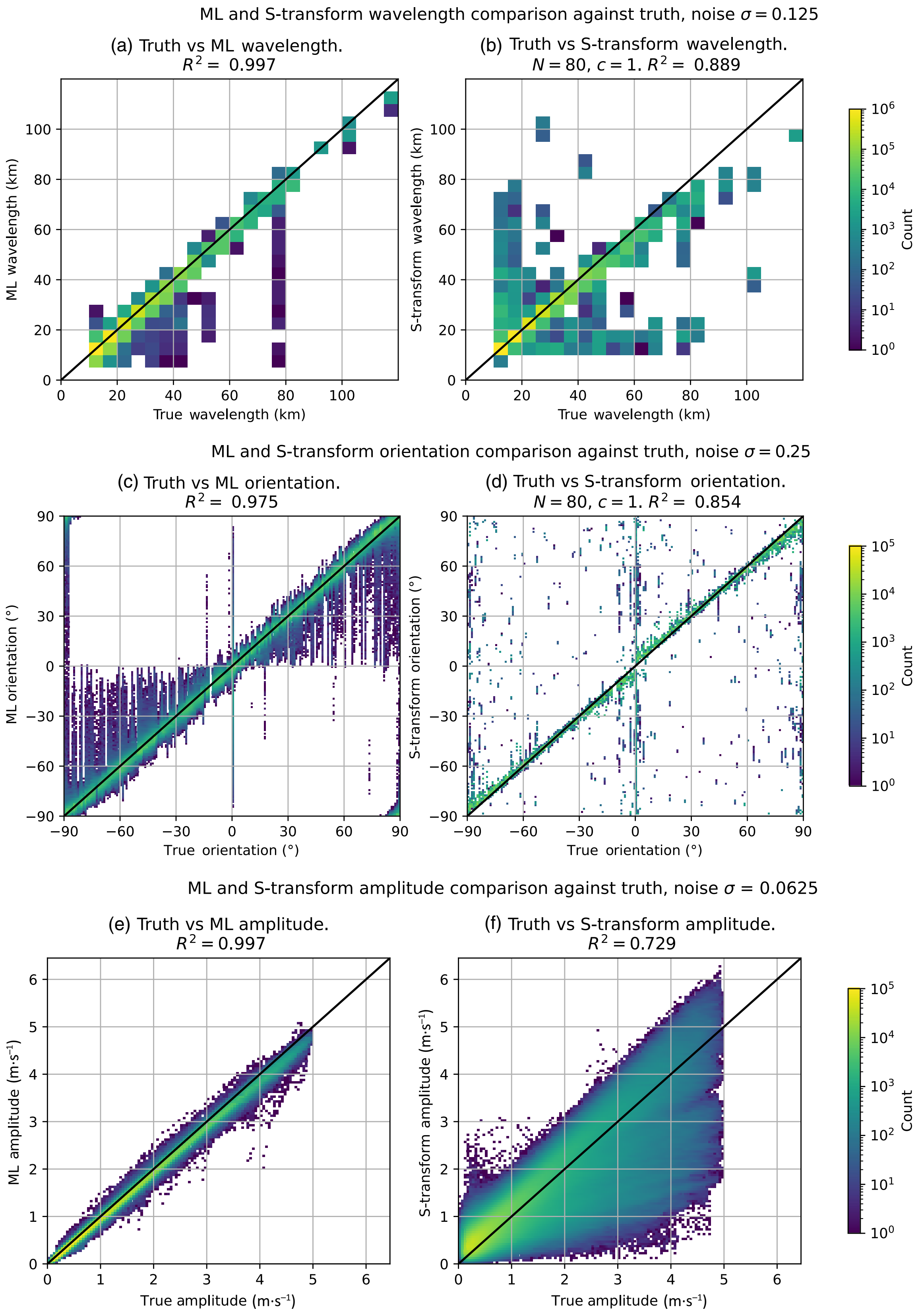

FIGURE 9 Synthetic data: comparison between true characteristics, machine learning (ML) model prediction, and S-transform characteristics for the synthetic test dataset (with 80 frequency voices and a scaling parameter $c = 1$). (a, c, e) Histogram of truth versus ML-derived characteristic from the test dataset. (b, d, f) Histogram of true characteristic versus S-transform from the test dataset. The black line in (a)–(f) is the line $y = x$. [Colour figure can be viewed at wileyonlinelibrary.com]

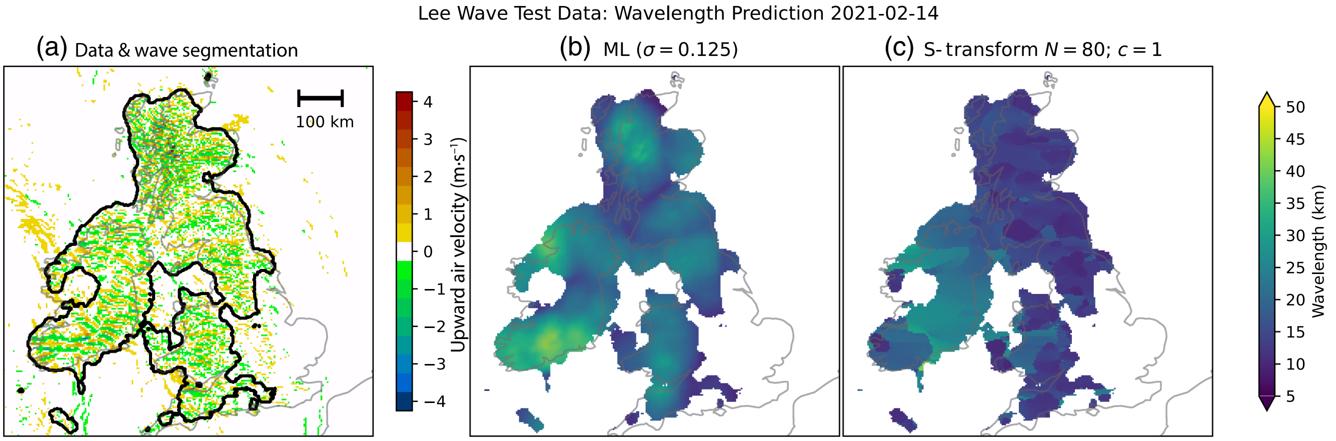
Lee Wave Test Data: Wavelength Prediction 2021-02-14
(a) Data & wave segmentation
100 km
4
3
2
1
0
−1
−2
−3
−4
Upward air velocity (m·s⁻¹)
(b) ML (σ = 0.125)
(c) S-transform N = 80; c = 1
50
45
40
35
30
25
20
15
10
5
Wavelength (km)

FIGURE 10 One selected example of machine-learned wavelengths against S-transform wavelength from UKV data, from February 14, 2021, 0900 UTC: (a) 700 hPa UKV vertical velocities and recognised lee wave regions; (b) WLMODEL (normally distributed noise, standard deviation $\sigma = 0.125$ in training data) derived wavelengths; (c) S-transform wavelength ($N = 80$ frequency voices, scaling parameter $c = 1$). [Colour figure can be viewed at wileyonlinelibrary.com]

ML wavelengths compare well against the synthetic wavelengths, with $R^2 = 0.996$. There is a high density of points along the $y = x$ line in Figure 9a. The S-transform-derived wavelengths compare with the truth favourably, though less so than the ML wavelengths, which is reflected in the lower R^2 value of 0.889 and the slightly larger scatter of points about the $y = x$ line in Figure 9b. The S-transform-derived wavelengths are too small at wavelengths greater than 80 km, which is not seen in the ML-derived wavelengths.

Figure 10 shows the S-transform and the ML-derived wavelengths for one example of UKV test data. The wavelengths predicted by the WLMODEL are reasonable, and relative wavelengths observed by eye correspond appropriately in both the ML prediction and the S-transform. For example in Figure 10 there is a region of longer wavelengths over the south of Ireland compared with shorter wavelengths over Scotland, which is predicted as such by the WLMODEL. The WLMODEL prediction shows a smoother field with greater variation in wavelength over the UK, but longer wavelengths than those produced by the S-transform. The S-transform wavelengths are, by comparison, more uniform compared with the ones from the WLMODEL. The sharp boundaries between regions in S-transform-derived wavelengths are a product of the S-transform reproducing a clean wave field. However, this results in unrealistic discontinuities in the S-transform wavelengths. For example in Figure 10c, there is a discontinuity in wavelength over the south of Ireland according to the S-transform that is not seen by eye in Figure 10a.

Figure 11 shows a histogram for the UKV February 2021 test set, comparing the WLMODEL predictions against the S-transform. The histogram only shows locations where the SEGMODEL predicts wave activity in the first place. Overall, the WLMODEL produces physically reasonable wavelengths (typical lee wave wavelengths are in the range 5–35 km; American Meteorological Society, 2012).

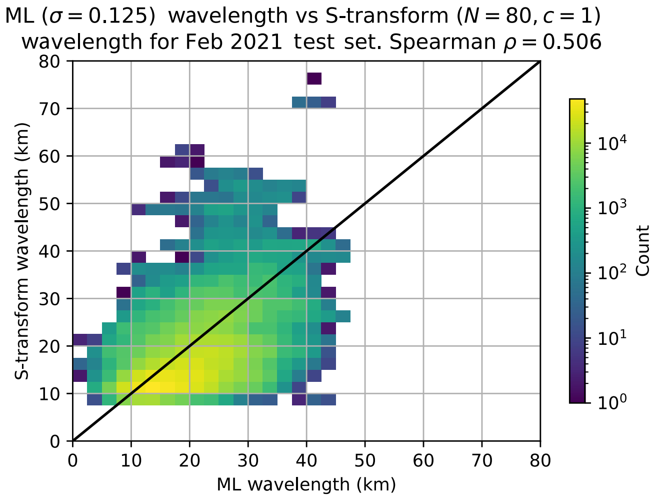

FIGURE 11 A histogram of the WLMODEL wavelengths against the S-transform wavelengths for the UKV test set. [Colour figure can be viewed at wileyonlinelibrary.com]

The WLMODEL wavelengths are slightly longer than the S-transform wavelengths. The range of wavelengths is smaller in the WLMODEL output than the S-transform. Overall, the WLMODEL gives reasonable indications of wavelength on the UKV data, compared with the S-transform. On the UKV data, whereas the WLMODEL wavelengths are typically longer than the wavelengths derived using the S-transform technique, the longest ML-derived wavelengths are less than 50 km compared

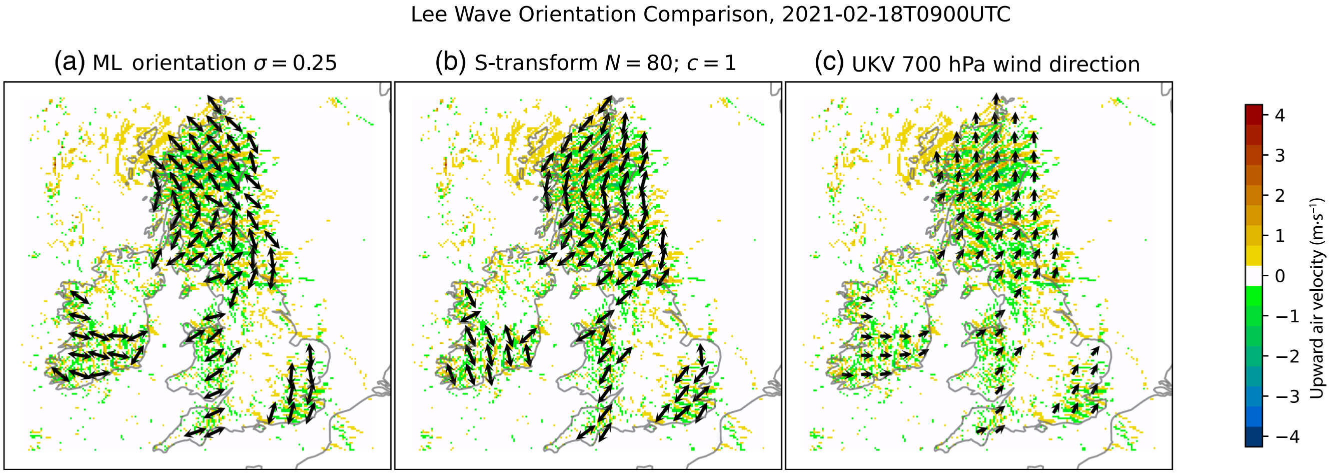
Lee Wave Orientation Comparison, 2021-02-18T0900UTC
(a) ML orientation $\sigma = 0.25$
(b) S-transform $N = 80$; $c = 1$
(c) UKV 700 hPa wind direction
4
3
2
1
0
−1
−2
−3
−4
Upward air velocity (m·s⁻¹)

FIGURE 12 Comparison of the ORIENTMODEL-predicted wave orientation, S-transform orientation, and UKV 700 hPa wind direction. The black arrows show the direction of wave propagation/wind direction and are therefore perpendicular to the wave-fronts. Wind direction is the closest variable in UKV to compare predicted wave orientation with. [Colour figure can be viewed at wileyonlinelibrary.com]

with the longest S-transform wavelengths being over 70 km. These longer S-transform wavelengths can occur in unrealistic locations (e.g., e one case occurs in a region less than 50 km in diameter). This is despite good correlation in the synthetic test data.

3.2.2 | Orientation

The angle predictions from the ORIENTMODEL and S-tranform were combined with the output from the original segmentation model, so that orientation predictions were only produced for regions containing waves. Figure 9c,d compares the performance of ORIENTMODEL ($\sigma = 0.25$) and S-transform (80 frequency voices and $c = 1$) derived orientations on the synthetic test set. Figure 9c shows that the S-transform-derived orientations compare well with the true orientation, with some seemingly random scatter across the axes. Figure 9d shows that the ML-derived orientations also compare well with the truth, but with greater spread towards 0°. Both methods of deriving orientation line up well along the $y = x$ line in each subplot.

In simple flows the wave crests would be expected to be perpendicular to the wind direction, and so wind direction can be used as a proxy for wave orientation. Owing to the three-dimensional nature of the orography and the fact that the waves are not monochromatic, this assumption is not perfectly true; however, the UKV wind direction at 700 hPa is still useful as an independent sanity check on the derived wave orientations.

Figure 12 shows an example of wave orientation (the ORIENTMODEL and S-transform) alongside UKV wind direction. ORIENTMODEL has done a good job of predicting the angle of the waves by eye. Neither tell the full picture, as the wind direction is not necessarily the same as the orientation of a wave. For example, regions with regular wave-like structures, such as those over Scotland, Wales, southeast England, and southwest England have plausible predicted orientations. However, waves with less structure, such as those over Ireland, have predicted orientations that are less convincing by eye. The S-transform in this case has not captured fully the change in orientation over Scotland and has the orientation more northerly than the ORIENTMODEL, which by eye seems to have captured the northwest–southeast orientation better.

Figure 13a shows the ORIENTMODEL orientation against the UKV wind direction for the test set. The data contain a high degree of scatter, though there is a relationship by eye between the ML orientation and wind direction. The least-squares correlation coefficient $R^2 = 0.116$ is low, however. Figure 13a does show that, in general, as the ORIENTMODEL orientations veer the wind direction veers as well, though not quite along the $y = x$ line. Figure 13b compares the S-transform orientation against the wind direction. This plot is also noisy, though by eye shows correlation against the wind direction. As already stated, the wind direction is not necessarily a good predictor for the wave orientation, and the fact that the data in Figure 13 do not follow a 1:1 line may actually be for a good physical reason; for example, because there is a preferential orientation of many of the mountain ranges over the UK. Finally, Figure 13c compares the ORIENTMODEL predictions against the S-transform. A positive trend is shown by eye, approximately along the $y = x$ line. The discontinuity in the S-transform-measured orientations around 0° and 90° is due to its formulation using

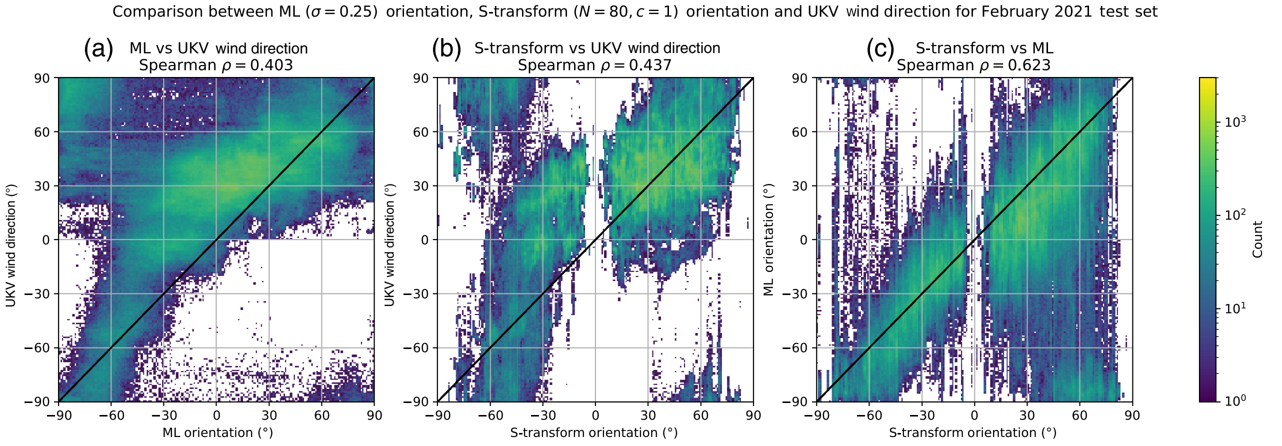

FIGURE 13 Histograms comparing the ORIENTMODEL orientation, S-transform orientation ($N = 80$, $c = 1$), and UKV wind direction at 700 hPa for the February 2021 test set in degrees from north (0°). In each case the line $y = x$ is plotted in black. [Colour figure can be viewed at wileyonlinelibrary.com]

the discrete fast Fourier transform. When orientated in exactly the x or y directions, wave numbers in the orthogonal direction are equal to zero, corresponding to an infinite wavelength, which is just the signal mean in that direction and is therefore not able to be localised. This is not a limitation for the ML approach. The data contain a high amount of scatter, but they do show some relationship between the ML-derived orientation and S-transform, suggesting that the output from the ORIENTMODEL is reasonable. However, it also suggests that deriving wave orientation is hard for both traditional spectral methods and ML methods.

3.2.3 | Amplitude

Figure 8c shows the wave amplitude predictions by the ML model for 0900 UTC on February 14, 2021. The largest amplitudes tend to be over hilly areas, such as in Scotland. The amplitude predictions produced are reasonable when compared by eye with the vertical velocities in Figure 8a. An alternative approach using pixelwise wavelengths to retrieve amplitudes by selecting the maximum vertical velocity within a region of the size of the wavelength for each pixel resulted in unrealistic large local variations in amplitude. This meant that there were large regions containing unreasonably large amplitudes. The neural network approach as described here produces amplitudes that are more consistent and smoothly varying over the length scale of the gravity wave envelopes seen visually in the vertical velocity data.

Figure 9e,f compares the ML and S-transform amplitudes with the true amplitudes for the synthetic data. The ML model performs well, with an $R^2 = 0.997$, whereas the S-transform amplitudes had $R^2 = 0.729$ compared with the truth. Figure 14 compares the AMPMODEL-derived amplitudes against the amplitudes from the S-transform. The amplitudes are well correlated, with a Spearman $\rho = 0.750$, and mainly focused around the $y = x$ line. At smaller amplitudes (≈0.5 m·s^{-1}), the ML model overestimates amplitude slightly compared with the S-transform, whereas at higher ML amplitudes (>2 m·s^{-1}) there is a higher spread of S-transform-predicted amplitudes.

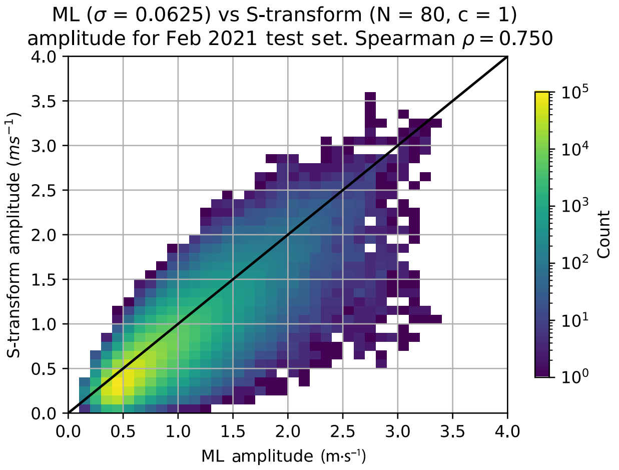

FIGURE 14 Histogram comparing the machine-learning (ML)-derived amplitudes against those from the S-transform for the February 2021 UKV test set. [Colour figure can be viewed at wileyonlinelibrary.com]

4 | CONCLUSIONS

A U-Net has been trained to identify regions of lee waves over Britain and Ireland from vertical velocity model data, and the final layers of the network have subsequently been fine-tuned separately to predict wavelength, orientation, and amplitude of the waves. Spectral techniques (e.g., the S-transform) do not permit the creation of a pixelwise wave mask as has been done here.

The trained segmentation model has a pixel accuracy of >95% when compared against hand-labelled truth. Despite being trained with a relatively modest training set of 335 scenes of vertical velocity data covering the whole UK at 2 km resolution, the SEGMODEL U-Net is skilful. The segmentation produced by the SEGMODEL is realistic, and in coherent regions; for example, those shown in Figure 7. The segmentation mask produced is on the same resolution as the NWP data, allowing precise localisation of wave forecasts. An accuracy closer to 100% would be difficult to obtain given the subjective nature of labelling waves near the edges of wave packets.

By using transfer learning, the copies of the trained SEGMODEL were fine-tuned on synthetic gravity wave data to estimate wave characteristics of gravity waves in UKV model output. The characteristics models still retain learned weights from being trained on the NWP data, with the synthetic data being used to extract characteristics from the model instead of a segmentation mask. Originally, these characteristics models were trained on data without noise, which, for the wavelength model, resulted in wavelengths being predicted on the NWP data that were too long. This has been rectified by training the characteristics models on noisy data.

On the UKV data, the wavelength model trained with noise tended to predict shorter wavelengths than with no noise, but longer than wavelengths derived using an S-transform. Though the noise used within the synthetic training data are unlikely to be the same as the background vertical velocities in the UKV data, these results do suggest that using noisy synthetic data might help make the ML models more robust to noise in real-world applications. How much noise to include in the training data, or how it should be distributed, remains an avenue to be explored, though using a small amount of noise such as $\sigma = 0.125$ or 0.25 seems reasonable given models trained on this magnitude of noise perform well across the different levels of noise in the synthetic data.

The orientation predictions, such as those shown in Figure 12, demonstrate that the ORIENTMODEL performs well at predicting the orientation of waves. Figure 12 shows that the wind direction at 700 hPa is not sufficient on its own to show the wave direction, so this method of deriving the wave orientation from the vertical velocities could add value to existing forecasts of waves and rotor activity. Figure 13 shows that the wind direction is less correlated with the ML ($\rho = 0.403$) and S-transform ($\rho = 0.437$) predictions compared with the S-transform and ML predictions ($\rho = 0.623$). On the UKV data, the models trained on noisy data still retain a smooth field, but with wavelengths and orientation closer to those from the S-transform. Several wavelength and orientation models could be run on the data to obtain a measure of the uncertainty between the ML model-derived characteristics.

AMPMODEL produces a smoothly varying prediction of the wave amplitude, where individual peaks in vertical velocity are smoothed out. The test set, despite being small, contains cases of large-amplitude waves (velocities in excess of 3 ms^{-1}; Vosper *et al.*, 2013), as demonstrated in Figure 8c. Large-amplitude waves have potential impacts downstream for the formation of rotors, and so successfully identifying these is important.

The models described here have been trained on and applied to two-dimensional vertical velocity NWP data over Britain and Ireland. Applying the same technique to other regions of the world would likely require retraining the models on the new data, as although the wave patterns would look similar, the other sources of vertical velocity in the data may not. In addition, this approach used two-dimensional slices of vertical velocities at 700 hPa, since lee waves are coherent in UKV model output over Britain and Ireland at 700 hPa. Using data on other pressure surfaces, or training a model on three-dimensional data, or with other variables such as vertical profiles of the Scorer parameter in addition to the vertical velocities, may produce models that are more skilful than the ones presented here.

These models provide an easy way to automatically derive information about waves from NWP model output. One benefit of using the segmentation and characteristics tools described is that the perceived severity of lee waves can be understood, as well as their generation in relation to the meteorology. One key advantage of the U-Net is that, once trained, it can be significantly cheaper and more efficient to run than the S-transform. For the 28 high spatial-resolution images in the February UKV test set used here, the U-Net can be up to 13 times faster than the S-transform to produce the wave mask, wavelength, and orientation measurements.

This efficiency, combined with their realistic output, makes these ML models a powerful new tool for post-processing NWP model output that can provide operational forecasters with an indication of where to expect trapped lee wave activity and the likely wave characteristics. These can aid forecasters when predicting high-impact wave and near-surface wind events.

The trained models are a potential benefit for forecasters in producing an automatic detection and characterisation of trapped lee waves directly from operational UKV model output. Another application of being able to automatically diagnose waves and their characteristics is the generation of a climatology of lee wave activity over Britain and Ireland from archived model output. Future work will include developing such a climatology of waves from NWP model output in order to learn more about the prevalence and characteristics of waves over the UK and to provide forecasters with guidance on the conditions under which strong lee wave events are likely to occur.

There are also avenues worth exploring within ML model development. For example, the effects of feeding additional data to the model, such as vertical velocities at different heights, surface winds, orography, or the Scorer parameter. Incorporating known physical relationships between different variables for lee waves into a deep-learning model may help a tool such as this one better diagnose waves and address some nervousness about using “black box” machine-learned models such as this operationally.

Though this work has used U-Nets to create an ML model capable of identifying and characterising lee waves, it also highlights the wider potential of these methods to be used in identifying a wide range of weather features and phenomena in high-resolution model data. The study also offers useful examples of leveraging the maximum impact from limited hand-labelled data by supplementing with augmentation and carefully constructed synthetic datasets. It is also a valuable example of how, with fine-tuning, an ML model developed to classify features can be used to identifying underlying physical characteristics of the features. Though these ideas could easily be applied to other wave problems in geophysical systems, they could equally be applied to a range of other types of feature in the atmosphere.

AUTHOR CONTRIBUTIONS

Jonathan Coney: conceptualization; formal analysis; investigation; methodology; visualization; writing – original draft. **Leif Denby:** conceptualization; funding acquisition; methodology; software; supervision; writing – review and editing. **Andrew N. Ross:** conceptualization; funding acquisition; methodology; supervision; writing – review and editing. **He Wang:** conceptualization; methodology; supervision; writing – review and editing. **Simon Vosper:** conceptualization; funding acquisition; methodology; supervision; writing – review and editing. **Annelize van Niekerk:** conceptualization; methodology; supervision; writing – review and editing. **Tom Dunstan:** conceptualization; methodology; supervision; writing – review and editing. **Neil Hindley:** software; writing – review and editing.

ACKNOWLEDGEMENTS

Thank you to the three anonymous reviewers for their time to read and provide feedback on the article. Thanks to Peter Sheridan, Steve Derbyshire, and the rest of the orography group at the Met Office for their knowledge of trapped lee waves, and their representation in Met Office model output. Thanks to Corwin Wright at the University of Bath for helpful suggestions about further applications of this work. This work used JASMIN, the UK collaborative data analysis facility. Computing work was undertaken in Python, making use of fastai v2.5.2 (Howard and Gugger, 2020) and PyTorch (Paszke *et al.*, 2019) (which fastai is built on) for training and interpreting deep-learning models. Jonathan Coney was supported in this work by the Leeds-York-Hull Natural Environment Research Council (NERC) Doctoral Training Partnership (DTP) Panorama under grant NE/S007458/1, and a CASE award from the Met Office. The article processing charge was covered by the UKRI block grant to the University of Leeds.

DATA AVAILABILITY STATEMENT

All the meteorological data were obtained from the Met Office Managed Archive Storage System (MASS), accessed through the JASMIN service. The 700 hPa vertical velocity data and segmentation masks used for training the segmentation model are available on Zenodo: Coney (2023b). Code to generate the synthetic data, the segmentation mask maker Jupyter notebook, and code to train and run models are available on GitHub: Denby (2023) and Coney (2023a). The S-transform analysis code can be found in Hindley (2021).

ORCID

Jonathan Coney https://orcid.org/0000-0001-7310-8002

REFERENCES

Ágústsson, H. & Ólafsson, H. (2014) Simulations of observed Lee waves and rotor turbulence. *Monthly Weather Review*, 142, 832–849.

American Meteorological Society. (2012) Mountain Wave. https://glossary.ametsoc.org/wiki/Mountain_wave

Coney, J. (2023a) Jdconey/leewavenet: initial release. 10.5281/zenodo.8193019.

Coney, J. (2023b) UKV Lee Waves 700 hPa vertical velocities and hand labels. https://doi.org/10.5281/zenodo.7565310

Darby, L.S. & Poulos, G.S. (2006) The evolution of lee-wave-rotor activity in the lee of Pike’s peak under the influence of a cold frontal passage: implications for aircraft safety. *Monthly Weather Review*, 134, 2857–2876.

Denby, L. (2023) Synthetic Gravity Waves. https://doi.org/10.5281/zenodo.7576811

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K. & Fei-Fei, L. (2009) ImageNet: a large-scale hierarchical image database. Paper presented at: 2009 IEEE conference on computer vision and pattern recognition, IEEE. 248–255 https://ieeexplore.ieee.org/document/5206848

Dev, S., Manandhar, S., Lee, Y.H. & Winkler, S. (2019) Multi-label cloud segmentation using a deep network. 2019 USNC-URSI radio science meeting (joint with AP-S symposium), USNC-URSI 2019-proceedings. 113–114.

Doyle, J.D. & Durran, D.R. (2002) The dynamics of mountain-wave-induced rotors. *Journal of the Atmospheric Sciences*, 59, 186–201.

Durran, D. (2003) Lee waves and mountain waves. In: Holton, J.R., Curry, J.A. & Pyle, J.A. (Eds.) *Encyclopedia of atmospheric sciences*. Burlington: Elsevier, pp. 1161–1169.

Elvidge, A.D., Vosper, S.B., Wells, H., Cheung, J.C.H., Derbyshire, S.H. & Turp, D. (2017) Moving towards a wave-resolved approach to forecasting mountain wave induced clear air turbulence. *Meteorological Applications*, 24, 540–550.

He, K., Zhang, X., Ren, S. & Sun, J. (2016) Deep residual learning for image recognition. Paper presented at: 2016 IEEE conference on computer vision and pattern recognition (CVPR), vol. 2016, IEEE. 770–778.

Hindley, N. (2021) nhindley/acp-2020-465: Analysis and figure code for acp journal article acp-2020-465 hindley *et al.* (2021). https://zenodo.org/record/4721883

Hindley, N.P., Smith, N.D., Wright, C.J., Rees, D.A.S. & Mitchell, N.J. (2016) A two-dimensional Stockwell transform for gravity wave analysis of AIRS measurements. *Atmospheric Measurement Techniques*, 9, 2545–2565.

Hindley, N.P., Wright, C.J., Gadian, A.M., Hoffmann, L., Hughes, J.K. & David, R. (2020) Stratospheric gravity-waves over the mountainous island of South Georgia: testing a high-resolution dynamical model with 3-D satellite observations and radiosondes. *Atmospheric Chemistry and Physics Discussions*, 2020, 1–50.

Hindley, N.P., Wright, C.J., Smith, N.D., Hoffmann, L., Holt, L.A. & Alexander, M.J. (2019) Gravity waves in the winter stratosphere over the Southern Ocean: high-resolution satellite observations and 3-D spectral analysis. *Atmospheric Chemistry and Physics*, 19, 15377–15414.

Hindley, N.P., Wright, C.J., Smith, N.D. & Mitchell, N.J. (2015) The southern stratospheric gravity wave hot spot: individual waves and their momentum fluxes measured by COSMIC GPS-RO. *Atmospheric Chemistry and Physics*, 15, 7797–7818.

Howard, J. & Gugger, S. (2020) Fastai: a layered API for deep learning. *Information*, 11, 108.

International Civil Aviation Organization. (1968) BOAC, Boeing 707, G-APFE, accident at the foot of Mount Fuji, Japan, on 5 march 1966. Tech. Rep. 7. United Kingdom: Board of Trade.

Jadon, S. (2020) A survey of loss functions for semantic segmentation. Paper presented at: 2020 IEEE conference on computational intelligence in bioinformatics and computational biology (CIBCB), IEEE. 1–7.

Matsuoka, D., Watanabe, S., Sato, K., Kawazoe, S., Yu, W. & Easterbrook, S. (2020) Application of deep learning to estimate atmospheric gravity wave parameters in reanalysis data sets. *Geophysical Research Letters*, 47, 1–10.

Mobbs, S., Vosper, S., Sheridan, P., Cardoso, R., Burton, R. & Arnold, S. (2005) Observations of downslope winds and rotors in the Falkland Islands. *Quarterly Journal of the Royal Meteorological Society*, 131, 329–351.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J. & Chanan, G. (2019) PyTorch: an imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, 32, 1–12.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B. & Grisel, O. (2011) Scikit-learn: machine learning in python. *Journal of Machine Learning Research*, 12, 2825–2830.

Ralph, F.M., Neiman, P.J., Keller, T.L., Levinson, D. & Fedor, L. (1997) Observations, simulations, and analysis of nonstationary trapped lee waves. *Journal of the Atmospheric Sciences*, 54, 1308–1333.

Ronneberger, O., Fischer, P. & Brox, T. (2015) U-net: convolutional networks for biomedical image segmentation. arXiv. http://arxiv.org/abs/1505.04597

Sheridan, P., Vosper, S. & Brown, P. (2017) Mountain waves in high resolution forecast models: automated diagnostics of wave severity and impact on surface winds. *Atmosphere*, 8, 24.

Shorten, C. & Khoshgoftaar, T.M. (2019) A survey on image data augmentation for deep learning. *Journal of Big Data*, 6, 60.

Stockwell, R., Mansinha, L. & Lowe, R. (1996) Localization of the complex spectrum: the S transform. *IEEE Transactions on Signal Processing*, 44, 998–1001.

Tang, Y., Lean, H.W. & Bornemann, J. (2013) The benefits of the met Office variable resolution NWP model for forecasting convection. *Meteorological Applications*, 20, 417–426.

Vosper, S.B., Sheridan, P.F. & Brown, A.R. (2006) Flow separation and rotor formation beneath two-dimensional trapped lee waves. *Quarterly Journal of the Royal Meteorological Society*, 132, 2415–2438.

Vosper, S.B., Wells, H., Sinclair, J.A. & Sheridan, P.F. (2013) A climatology of lee waves over the UK derived from model forecasts. *Meteorological Applications*, 20, 466–481.

Wright, C.J., Hindley, N.P., Hoffmann, L., Alexander, M.J. & Mitchell, N.J. (2017) Exploring gravity wave characteristics in 3-D using a novel S-transform technique: AIRS/aqua measurements over the southern Andes and Drake Passage. *Atmospheric Chemistry and Physics*, 17, 8553–8575.

Zhang, P., Ke, Y., Zhang, Z., Wang, M., Li, P. & Zhang, S. (2018) Urban land use and land cover classification using novel deep learning models based on high spatial resolution satellite imagery. *Sensors*, 18, 3717.

How to cite this article: Coney, J., Denby, L., Ross, A.N., Wang, H., Vosper, S., van Niekerk, A. *et al.* (2024) Identifying and characterising trapped lee waves using deep learning techniques. *Quarterly Journal of the Royal Meteorological Society*, 150(758), 213–231. Available from: https://doi.org/10.1002/qj.4592