Research Article Vol. 30, No. 4 / 14 Feb 2022 / Optics Express 5540
Optics EXPRESS

Coherent Doppler wind lidar with real-time wind processing and low signal-to-noise ratio reconstruction based on a convolutional neural network

OLIVER KLIEBISCH,[1,*] HUGO UITTENBOSCH,[1] JOHANN THURN,[1] AND PETER MAHNKE[1]

[1] *Institute of Technical Physics, German Aerospace Center, Pfaffenwaldring 38-40, 70569 Stuttgart, Germany*
[*] *oliver.kliebisch@dlr.de*

Abstract: Multi-classification using a convolutional neural network (CNN) is proposed as a denoising method for coherent Doppler wind lidar (CDWL) data. The method is intended to enhance the usable range of a CDWL beyond the atmospheric boundary layer (ABL). The method is implemented and tested in an all-fiber pulsed CWDL system operating at 1550 nm wavelength with 20 kHz repetition rate, 300 ns pulse length and 180 µJ of laser energy. Real-time pre-processing using a field programmable gate array (FPGA) is implemented producing averaged lidar spectrograms. Real-world measurement data is labeled using conventional frequency estimators and mixed with simulated spectrograms for training of the CNN. First results of this methods show that the CNN can outperform conventional frequency estimations substantially in terms of maximum range and delivers reasonable output in very low signal-to-noise (SNR) situations while still delivering accurate results in the high-SNR regime. Comparing the CNN output with radiosonde data shows the feasibility of the proposed method.

1. Introduction

The remote sensing of wind velocity using lidar systems is an established technique with applications in atmospheric research [1], wind energy systems [2] and airport turbulence and wind shear monitoring [3]. Apart from ground based installations, airborne [4,5] and even spaceborne [6] wind lidar systems have also been demonstrated. Wind lidar systems are either based on a direct detection of photons scattered from the atmosphere using a spectral analyzer for Doppler shift discrimination or on coherent detection where the received photons are mixed with a local oscillator and the resulting beat signal is analyzed. In this paper we focus on the latter lidar principle. Coherent detection lidar relies on the presence of aerosols (Mie scatterers) resulting in a narrow-band Doppler-shifted backscattering signal. The dependence on aerosol concentration is challenging as the signal quality and maximum range is highly variable with local meteorologic conditions.

A typical feature of a ground-based Doppler wind lidar is a steep drop in SNR after the ABL due to a reduction of the aerosol concentration of 1 to 3 orders of magnitude.

While there exists algorithms in different disciplines for spectral peak detection like discrete spectral peak estimation [7], moment velocity estimators [8], continuous wavelet transform-based pattern matching [9] and many more [10], these are limited by the information that can be extracted per range gate and will start to perform poorly in very low-SNR regimes.

In this work we suggest using a convolutional neural network trained on real and simulated coherent lidar signals in order to achieve a robust estimate of the Doppler frequency shift

#445287 https://doi.org/10.1364/OE.445287
Journal © 2022 Received 14 Oct 2021; revised 15 Dec 2021; accepted 16 Dec 2021; published 7 Feb 2022

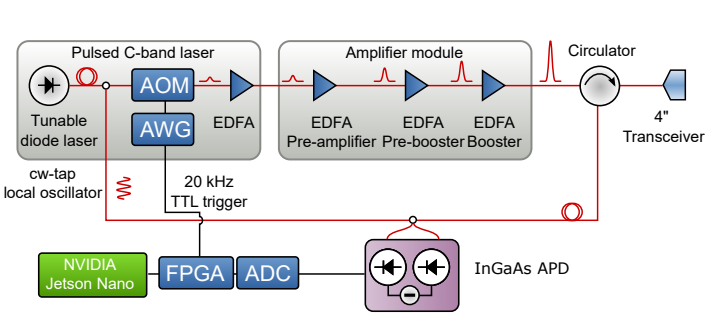
Optics EXPRESS

and therefore the line-of-sight wind speed by analyzing multiple range gates at once and thus extracting additional information about the lidar spectrum.

The network is used as a multi-classifier instead of a regressor which might seem counter-intuitive at first, but the reasoning will be made clear in the following sections. A detailed description of the training data generation and application of the network is presented. Lidar measurements have been performed using a self-built all-fiber coherent Doppler wind lidar operating at 1550 nm wavelength. We also present an FPGA-based real-time wind signal processing scheme which reduces the data bandwidth substantially and renders quick-looks unnecessary. The measured lidar data is labeled automatically using a spectral centroid frequency estimator and mixed with simulated lidar spectra as training and test data for the neural network. Via a simple comparison with radiosonde data launched in the vicinity of our lidar measurement site, we have verified our data evaluation approach.

2. Coherent Doppler wind lidar setup

The transmitter of our lidar system is a commercial off-the-shelf Erbium:fiber master-oscillator power amplifier system (Lumibird PEFL-EOLA). A summary of all system parameters is given in Tab. 1 and a schematic is depicted in Fig. 1. The lidar is an all-fiber system using polarization-maintaining (PM) fibers, except for the output fiber, which is a large-mode-area (LMA) fiber for handling of high pulse powers.



(a)

(b)

Fig. 1. (a) Schematic of the wind lidar setup. The laser system consists of the main oscillator unit generating the pulses, a cw seed tap and a three-stage power amplifier. The mono-static design is completed using a fiber circulator and a transmit/receive (transceiver) optics system. A detailed description is given in the text. **AOM**: acousto-optic modulator, **AWG**: arbitrary waveform generator, **EDFA**: erbium-doped fiber amplifier, **APD** avalanche photodiode, **ADC**: analog-to-digital converter, **FPGA**: field-programmable gate array. (b) Picture of the full lidar system as a 19-inch rack-mounted system and a double mount tripod. The booster amplifier and the transceiver optics are both attached to the tripod. This leads to a short fiber connection between the EDFA and the transceiver.

The laser system consists of a low-power pulsed laser and a multi-stage amplifier. The laser pulse is generated by chopping the output of a tunable diode laser at 1548.1 nm into 300 ns long pulses using an acousto-optic modulator which also induces a frequency shift f_{offset} of 80 MHz. The pulse shape is controlled using an arbitrary waveform generator which is triggered by the FPGA at 20 kHz rate. The pulse trigger as well as the timing synchronization is described in more detail in section 3.. The first erbium-doped fiber amplifier (EDFA) amplifies the pulse energy to around 0.6 µJ. The output pulse is sent through a transfer fiber to the power amplifier EDFA which is mounted with the transceiver optics on a double mount tripod platform. The

Optics EXPRESS

Table 1. Lidar system parameters.

Parameter	Value
Pulse energy	180 µJ
Pulse length	300 ns
Repetition rate	20 kHz
Laser wavelength	1548.1 nm
Linewidth (cw)	5 kHz
Local oscillator offset	80 MHz
Telescope diameter	100 mm
Focal length	500 mm
Beam quality M^2	1.5
Detector bandwidth	400 MHz
ADC sampling rate	250 megasamples /s
ADC bit depth	16 bits
Duty cycle	98.3 %

power amplifier consists of three EDFA stages, named pre-amplifier, pre-booster and booster amplifier. An internal fiber circulator is used to split the received from the transmitted light. At the output of the circulator up to 180 µJ of pulse energy is provided in a LMA fiber with 25 µm core diameter.

A monostatic lidar configuration is used with a single aspheric lens with 500 mm focal length and 100 mm diameter. The lens is mounted in a barrel attached to the second tripod mount point. An adjustable FC/APC fiber connector is mounted at the end of the barrel for the alignment of the collimated beam. A 9-axis inertial measurement unit (IMU, Bosch BNO055) is used to determine the attitude of the transceiver. The backscattered light is split of at the return port of the fiber circulator, sent to a 2×2 fiber coupler and brought to interference with a cw tap of the diode laser in the master oscillator. The outputs of the fiber coupler are connected to a balanced avalanche photodiode detector (Thorlabs PDB570C). The transimpedance gain is 1 kV/A and the avalanche multiplication factor is set to $M = 10$. The photodetector output signal is subsequently digitized, pre-processed in real-time by the FPGA and then transferred to a single board computer (NVIDIA Jetson Nano) where post-processing, data storage and laser control is handled.

The full system is integrated into a 19-inch 3U rack module. A keyboard-video-mouse console is used for system operation as well as power distribution in the system. The whole system uses a single 24 V power supply and has a power consumption of about 150 Watt. A movable telescope tripod with two mounting positions is used for the transceiver and the power amplifier EDFA module. This design requires an electrical and fiber-optic interface to the tripod but at the same time a very short LMA-fiber of about 30 cm can be used to connect the laser output to the transceiver. This reduces any potential stimulated Brillouin scattering problems arising from the high pulse energy in the fibers. A photograph of the system during operation is shown in Fig. ??.

3. FPGA-based real-time processing

Coherent Doppler wind lidar systems usually require high speed digitizers with sampling rates exceeding 100 megasamples/s. This leads to a high raw data bandwidth and imposes high requirements to the digital processing hardware. Depending on the processing power, typical strategies involve either raw streaming of the data to a storage device with quick-look processing for live monitoring or a direct CPU/GPU-accelerated data processing. More recent developments

Optics EXPRESS

have used FPGAs to accelerate the data processing [11,12] of coherent Doppler wind lidar systems.

In the presented lidar system we have implemented the division of the lidar return signal into range gates with subsequent windowed fast Fourier transform (FFT) and incoherent averaging of the periodograms. The processing scheme is illustrated in Fig. 2. The whole implementation is based on a Xilinx Zynq 7020 system-on-a-chip FPGA and an FPGA-mezzanine card with the ADC chip (Analog Devices AD9467). The FPGA is mounted on a Trenz TE0701 carrier board. The following logic design is based on [13] and has been adapted for the use with a coherent Doppler wind lidar. Data is streamed from the ADC to the FPGA using parallel low-voltage differential signaling (LVDS). For stable averaging of the acquired lidar signal the ADC, FPGA and laser pulse have to be synchronized to a common reference. We use an Analog Devices AD9517 clock distribution chip which stabilizes its voltage-controlled oscillator to an external 125 MHz quartz oscillator using a phase-locked loop. The clock distribution chip generates a 250 MHz signal which is used by the ADC and the FPGA for synchronization of the logic gates to the ADC clock. The logic design uses the ADC clock to derive a synchronous 20 kHz trigger signal for the pulsed laser. The optical pulse jitter has been measured using a photodetector and an oscilloscope and shows a normal distribution with a full-width at half maximum on the order of 1 clock cycle. The measured value is 4.94 ns. Additionally, the latency between the falling edge and optical pulse emission is 3.6625 µs. The trigger signal is also used to periodically reset and re-arm the processing logic. A programmable delay can be set in order to compensate the latency between falling edge transition and pulse emission.

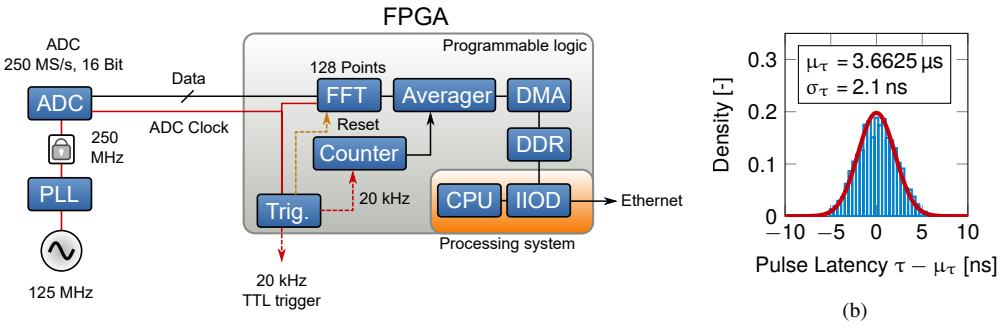


Fig. 2. (a) Schematic of the data processing scheme using a Xilinx Zynq 7020 FPGA and a 250 megasamples\s ADC. The FPGA handles the laser pulse trigger generation as well as synchronous range-gate division and power spectrum calculation in real time. **ADC**: analog-to-digital converter, **PLL**: phase-locked loop, **FFT**: fast Fourier transform, **DMA**: direct memory access, **DDR**: double data rate (RAM), **CPU**: central processing unit (dual core ARM A9 in this case), **IIOD**: industrial I/O daemon. (b) Histogram of the pulse latency of 2017 laser pulses with respect to the 20 kHz TTL-trigger signal for the determination of the pulse jitter.

The raw data is multiplied with a programmable window function and then fed into a continuous pipelined FFT [14] with 128 points. This defines the effective range gate length as ≈ 77 m. From the complex output of the FFT a power spectrum estimate is determined by calculating the absolute squared value. The power spectrum output of each 128 samples deep range gate is accumulated in a block RAM memory and added to the data from previous pulses, effectively averaging the power spectra incoherently (i.e. ignoring any phase information in the data). The number of averages is fixed to $2^{14} = 16384$ pulses, resulting in an integration time of precisely 0.8192 s. However, due to the non-integer ratio between the number of range gates per pulse and

Optics EXPRESS

short latencies within the FPGA, the time window of 50 µs can not entirely be considered for data averaging and the last two range gates have to be discarded. The remaining 96 range gates correspond to a duty cycle of the data processing of 98.3 %.

The averaged power spectra per range gate are transferred to the DDR RAM of the FPGA using a direct memory access controller. The processing system of the FPGA runs a customized Petalinux distribution and uses the industrial I/O kernel subsystem to transfer the data to a NVIDIA Jetson Nano using Gigabit Ethernet. The single board computer runs a measurement and control GUI written in Python. This pre-processing scheme results in a data bandwidth reduction from 500 MByte/s down to only 60 kByte/s.

4. Convolutional neural network based wind speed retrieval

The prediction of wind field data using artificial neural networks for Doppler wind lidars either for wind energy applications [15] or even in the ABL has been demonstrated [16]. However, these applications work on the final wind speed output of a wind lidar and attempt to perform temporal and spatial predictions of the wind speeds in the near future.

Instead, our approach focuses on the improvement of extracting Doppler shift values from the raw data at very low signal-to-noise ratios, where conventional frequency estimators are prone to instability and wrong results. In Fig. 3(a) the relevant part of a single raw lidar spectrogram is shown. The spectrogram shows the line-of-sight wind speed on the x-axis and the range gate distance on the y-axis. The wind speed is calculated from the Doppler frequency shift, where due to the acousto-optic modulator, a speed of 0 m/s corresponds to a signal frequency of 80 MHz. The wind speed is displayed here for simplicity. The lidar formulas in this text refer to the Doppler frequency bin instead of the line-of-sight wind speed.

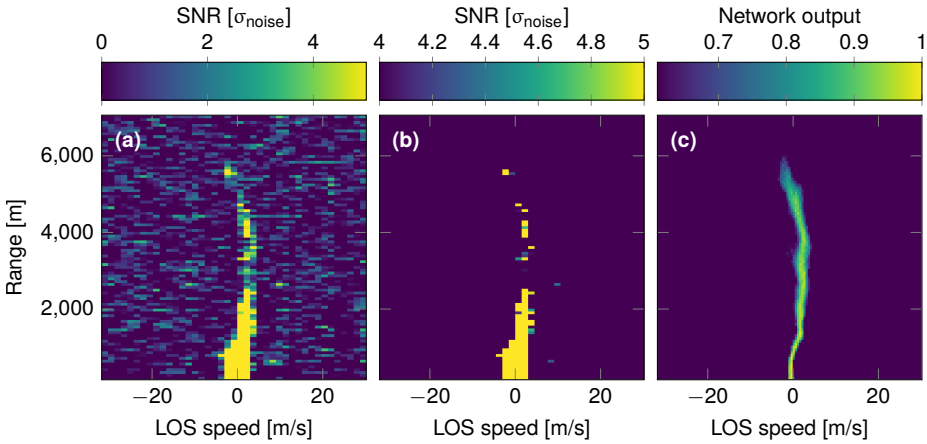


Fig. 3. Different visualizations of a single measured spectrogram and comparison with the output of the trained convolutional neural network explained in this work. All three figures show the line-of-sight speed on the x axis and the range on the y axis. (a) Single raw spectrogram as calculated by the FPGA using $2^{14} = 16384$ individual laser shots. The color of each pixel is scaled between a SNR of zero and $5\sigma_{\text{noise}}$. After about 2500 m a faint transition into the noise is apparent. (b) Identical to (a) but the colorbar is scaled between $4\sigma_{\text{noise}}$ and $5\sigma_{\text{noise}}$ which highlights the working range of a regular spectral centroid algorithm. (c) Output of the neural network given the signal in (a) where the neuron activation has been normalized to the range of 0 to 1. A threshold of 0.65 has been applied.

The main noise source of the coherent wind lidar is the shot noise of the balanced detector which is a normally distributed noise source [17]. In the power spectrum this noise gets transformed to

Optics EXPRESS

an exponentially distributed - in our discrete case Poisson distributed - power spectral noise with an expectation value of the shot noise power spectral density $p_{\mathrm{shot}} = 2eI_{\mathrm{photo}}$ in A2/Hz. The standard deviation of the power spectral density is similar to the expectation value. By averaging N power spectra the noise distribution converges to a normal distribution with an expectation value of p_{shot} and a standard deviation $\sigma_{\mathrm{noise}} = p_{\mathrm{shot}} \Big/ \sqrt{N}$. To take the narrowing of the standard deviation into account, we use an offset corrected definition (called standardization in this work in accordance to typical pre-processing approaches used for efficient back-propagation in neural networks [18]) of the signal to noise ratio:

$$\mathrm{SNR} = \frac{p_{\mathrm{signal}} - p_{\mathrm{shot}}}{\sigma_{\mathrm{noise}}} = \sqrt{N}\left(\frac{p_{\mathrm{signal}} - p_{\mathrm{shot}}}{p_{\mathrm{shot}}}\right). \tag{1}$$

Therefore the signal features are expressed in multiples of the standard deviation σ_{noise} and it can be seen that the SNR increases with the number of averages as $\sqrt{N}$.

One way to determine the beat signal frequency in each range gate is to use an interpolated spectral centroid algorithm [19,20]. To avoid the evaluation of range gates with too low SNR a threshold is applied. The minimum signal in a range gate is therefore $S_{\mathrm{min}} = n_\sigma \cdot \sigma_{\mathrm{noise}}$. This value has to be chosen as a good compromise between false trigger rate and the number of missed weak signals. For perfectly white noise the chance for the spectral centroid evaluating a range gate containing only noise is $1 - (1 - 1/15787)^{41} = 0.26\%$ for $n_\sigma = 4$. This situation is visualized in Fig. 3(b) showing the remaining signal using thresholding. The noise floor of the lidar spectrogram is the result of averaging the exponential-type noise distribution of the power spectrum and residual contributions from external electrical noise sources. Therefore the distribution of the noise floor is not perfectly normally distributed.

Another problem with statistical thresholding is apparent to the naked eye. Only the information within each range gate is used while the data indicates that range gates with spectral information with lower peak SNR could still be extracted. However, relying on thresholding leads either to missing weak signals or too many false triggers degrading the evaluation result with false data. Our proposed solution to this problem is to apply approaches from image processing using neural networks onto a lidar spectrogram. As the neural network architecture, an adaption of the SqueezeNet [21] deep convolutional neural network has been chosen due to the good compromise between model accuracy, number of model parameters and required training resources. This might also enable the integration of the trained network into the FPGA pre-processing scheme in the future. Different network architectures have been implemented as well, but a full comparison is beyond the scope of this paper. The network has been implemented using the *Julia* programming language [22] with its *Flux.jl* [23] machine learning framework. The network architecture is shown in Fig. 4. We have adapted the SqueezeNet architecture to work on our input data consisting of matrices with dimension of 93 range gates $\times$ 41 frequency bins with a single channel instead of three in the original implementation. An exemplary result of applying the trained network to the raw signal shown in Fig. 3(a) is shown in Fig. 3(c). It should be stressed that the width of the CNN output no longer correlates with the spectral width of the raw signal. This is due to the choice of a multi-classification network instead of a regressor type network. The broadening of the neural network output for the higher ranges just indiciates the increasing uncertainty of network in identifying the peak frequency bin for low signal to noise ratios.

Training and test data is generated in the following way: Real measurement data is evaluated using the spectral centroid algorithm explained in this section. All range gate bins without an evaluable signal are replaced with artificial white noise given by a normal distribution with mean $\mu = 0$ and standard deviation σ_{noise}, i.e. $\mathcal{N}(0, \sigma_{\mathrm{noise}})$. The data is cropped and only data between the range gate bins $n_{r,\mathrm{min}}$ and $n_{r,\mathrm{max}}$ as well as the frequency bins $n_{f,\mathrm{min}}$ and $n_{f,\mathrm{max}}$ is used. The frequency axis can be interpolated by an integer factor m. Throughout this paper the interpolation

Optics EXPRESS

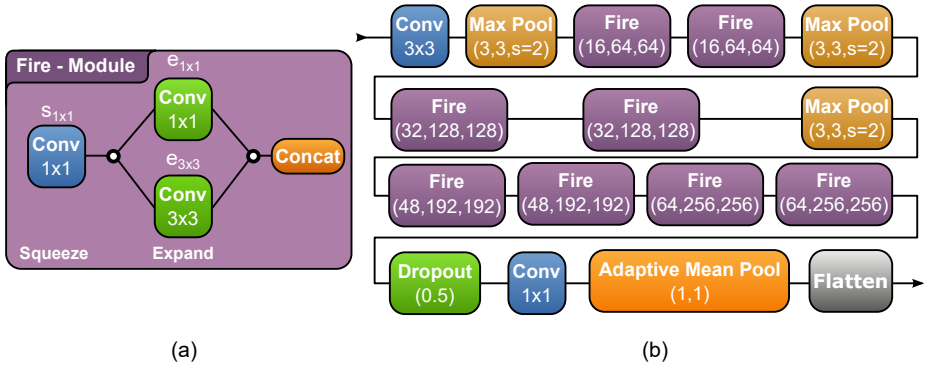


Fig. 4. Simplified schematic view of the neural network architecture chosen for this work. Full layer sizes are omitted for the sake of compactness. The architecture is basically a SqueezeNet 1.1 architecture. The left side (a) shows the core component of the network, the fire module. This module is parametrized using the size of the squeeze convolution layers (Conv layer) $s_{1\times1}$ and the size of the expansion layers $e_{1\times1}$ and $e_{3\times3}$. The full network architecture is shown on the right (b) where multiple fire modules are chained with max pooling layers in between. The parameter s for the max pooling layers is the stride. The adaptive mean pool layer is an averaging pooling layer which adapts ins size automatically to the given input and output size.

of $m = 3$ is chosen. The frequency bin determined by the spectral centroid is used as a label using one-hot encoding for each range gate, therefore representing a multi-classification type problem.

Using the real measurement data, the network cannot learn "new" features in the data as no ground truth is known and the network is basically trained to deliver the same results as the spectral centroid method. To cope with this, a training data generator has been implemented which simulates lidar spectrograms that mimic the signal features found in real signals. The signal generator uses partially long established lidar equations as well as simplified, semi-empirical formulas to add features like clouds. Most parameters are modeled as normally distributed, where the mean and standard deviation is chosen carefully to cover the range of experimentally observed values. The following basic lidar equation [24] is used to generate a simulated spectrogram:

$$S(R,f) = \frac{A_0\eta_A(R)\beta(R)}{\sqrt{2\pi}\sigma_{\mathrm{D}}(R)}\frac{1}{R^2}\cdot\exp\left(-2\int_0^R\sigma(r)\,\mathrm{d}r\right)\cdot\exp\left(-\frac{(f-\Delta f_{\mathrm{D}}(R))^2}{2\sigma_{\mathrm{D}}^2(R)}\right)+\mathcal{N}(0,\sigma_{\mathrm{noise}}). \quad (2)$$

R is the range gate distance, f the bin frequency and A_0 is an amplitude factor where all technical system parameters (telescope aperture, detector gain, responsivity,. . .) are summarized. $\Delta f_{\mathrm{D}}(R) = 2/\lambda_{\mathrm{laser}}v_{\mathrm{LOS}}(R)$ is the Doppler frequency shift for the line-of-sight velocity $v_{\mathrm{LOS}}(R)$ with standard deviation $\sigma_{\mathrm{D}}(R)$ which is the convolution of the pulsed laser linewidth and the wind speed distribution. For the simulation the range gate number n is used for indexing. The range is then given as $R = n\cdot\Delta R$, with ΔR as the range gate length. A random-walk is chosen for simulating the range-dependent Doppler shift as a series of the form

$$\Delta f_{\mathrm{D},n}(R = n\cdot\Delta R) = \begin{cases}\mathcal{N}(\Delta f_0,\sigma_{\Delta f_0}) & n = 1\\ \Delta f_{\mathrm{D},n-1}+\mathcal{N}(0,\sigma_{\Delta f}) & n>1\end{cases}, \quad (3)$$

with Δf_0 as the mean initial Doppler frequency shift and standard deviation $\sigma_{\Delta f_0}$.

Optics EXPRESS

The term $\eta_A(R)$ is the antenna function, which is given for the monostatic collimated lidar with Rayleigh range z_R of the laser beam as [24]:

$$\eta_A(R) = \frac{1}{2 + 2\left(\frac{z_R}{R}\right)^2}. \tag{4}$$

The remaining quantities of Eq. (2) are the volumetric backscattering coefficient $\beta(R)$ and the extinction $\sigma(R)$, where for simplicity it is assumed that $\beta(R) = P(-\pi)/4\pi\sigma(R)$ with an ensemble averaged backscattering phase $P(-\pi)$.

The backscattering coefficient is simulated as a series, similar to the Doppler shift:

$$\beta_n(R = n \cdot \Delta R) = \begin{cases} \mathcal{N}(\hat{\beta}_0, \sigma_{\hat{\beta}_0}) & n = 1 \\ \beta_{n-1} \cdot (1 + \mathcal{N}(0, \sigma_\beta)) & 1<n<n_{\mathrm{ABL}} \\ \frac{\beta_{n-1} \cdot (1+\mathcal{N}(0,\sigma_\beta))}{\beta_{\mathrm{decay}}} & n \geq n_{\mathrm{ABL}} \end{cases}. \tag{5}$$

The initial backscattering has the mean $\hat{\beta}_0$ with standard deviation $\sigma_{\hat{\beta}_0}$ and subsequent range gates fluctuate with a standard deviation of σ_β. After the ABL height $n_{\mathrm{ABL}} = \mathcal{N}(n_{\mathrm{ABL}_0}, \sigma_{n_{\mathrm{ABL}_0}})$ is reached, the backscattering is decaying by the factor β_{decay}. Clouds are included in the model by modification of the calculated series β_n. First, the number of clouds in the signal is randomly chosen on the uniform integer interval of $[0, m_{\mathrm{cloud,max}}]$. In the following, uniform integer distributions on the interval $[A, B]$ will be denoted with $\mathcal{U}_{[A,B]}$. In the next step the cloud center range gate $n_{\mathrm{cloud,center}}$, the cloud width $\Delta n_{\mathrm{cloud}}$ and the cloud scattering amplitude β_{cloud} are calculated by:

$$n_{\mathrm{cloud,center}} = \mathcal{U}_{[n_{\mathrm{cloud,min}}, n_{\mathrm{cloud,max}}]}, \tag{6}$$

$$\Delta n_{\mathrm{cloud}} = \left\lceil \left| \mathcal{N}(\Delta n_{\mathrm{cloud},0}, \sigma_{\Delta_{n_{\mathrm{cloud},0}}}) \right| \right\rceil, \tag{7}$$

$$\beta_{\mathrm{cloud}} = 10^{\mathcal{N}(\beta_{\mathrm{cloud},0}, \sigma_{\beta_{\mathrm{cloud},0}})}. \tag{8}$$

These parameters are used to set the backscattering coefficient and the extinction accordingly for the affected range gates. After calculating the simulated signal, the one-hot encoding matrix is derived. Only range gates with a signal of an amplitude larger than σ_{noise} are labeled. This allows the network to learn that range gates do not have to contain a classifiable signal. The set of parameters used in this publication can be found in the supplementary document. In general the values (especially the order of magnitude) can be motivated from the physical system parameter and backscattering properties of the atmosphere. However, in practice the parameter A_0 and the backscattering properties have been manually tuned to get simulated signals which are similar to the measured data.

The network is trained using backpropagation with a learning rate of 1×10^{-4} to 1×10^{-3} with an ADAM [25] optimizer. As a loss function the logistic cross-entropy is chosen. 1 000 000 training data sets containing 50 000 labeled real measurements and 50 000 test data sets containing 5000 real measurements are used. This corresponds to a 20 : 1 training to test data split. Typically a good fit of the model is achieved after around 25 to 50 epochs. The training time using CUDA-accelerated GPU training is about 12 minutes per epoch on a NVIDIA RTX 4000.

For demonstration of the proposed method, a comparison of the spectral centroid and the neural network is shown in Fig. 5. A synthetic spectrogram has been calculated using Eq. (2) and evaluated using both methods. The ground truth for the Doppler shift has been recorded and converted to line-of-sight velocity. The ground truth data has been limited at an SNR of 1σ, therefore only about 4000 m of range is visible. The spectral centroid algorithm has been only applied to data where the SNR is at least 4σ and therefore ends at a lower range compared to the neural network. The neural network has been applied for a normalized neuron activation of at

Optics EXPRESS

least 75 % which covers the whole ground truth range in this case. This is no coincidence, as the network is trained to classify data only to a SNR limit of 1σ. For lower signals the neuron activation will significantly drop. This can also be seen in Fig. 3(c).

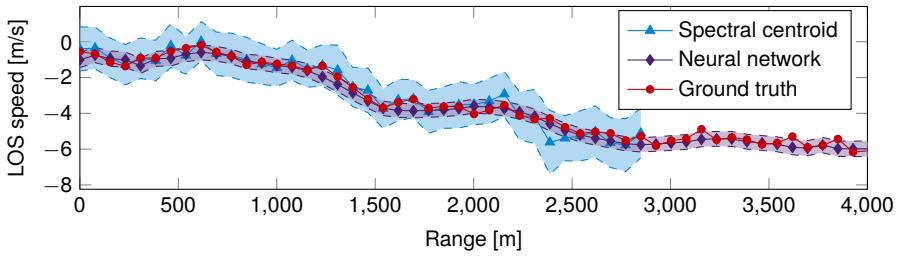


Fig. 5. Exemplary evaluation result of a synthetic lidar signal comparing the spectral centroid method with the proposed neural network evaluation method. The ground truth wind speed is drawn as connected red dots. The spectral centroid result is shown as connected blue triangles and the filled area marks the uncertainty of this algorithm. Likewise, the neural network result is shown as connected purple diamonds and filled area for the corresponding uncertainty. The coefficient of determination R^2 is 0.956 for the spectral centroid method and 0.978 for the neural network method.

This depiction shows a case where the neural network has a superior performance compared to the spectral centroid method. The uncertainty is lower for the neural network compared to the spectral centroid and the neural network has a lower deviation from the ground truth especially in the region of lower SNR at a range of about 2000 m and above. Furthermore, the neural network shows an accurate output close to the ground truth at ranges where the spectral centroid method fails to produce an estimate. We would like to stress, that this is just an exemplary demonstration of a spectrogram where the neural network shows superior performance and cannot be trivially generalized.

5. Experimental results

Lidar measurements during daytime have been performed over several months in order to test the system and to generate training data using the already presented approach. In Fig. 6 an evaluated lidar measurement for July 27$^{\text{th}}$, 2021 is shown. The LOS was oriented in western direction with an elevation angle of 45°. This measurement has been chosen because of changing weather conditions with respect to wind speeds and cloud formation. In Fig. 6(a) the LOS wind speed has been estimated using the spectral centroid estimator, showing a good performance to about 2000 m of range until noon and up to 3000 m in the late afternoon showing an increasing ABL height. Additionally, clouds occasionally passed the LOS. There is also range gate aliasing visible where the cloud distance exceeds the maximum range of 7.5 km and thus the signal reappears in the lower range gates. Due to the high SNR of clouds, the spectral centroid algorithm cannot discriminate between clouds and clear-air aerosol signals in this type of implementation.

In Fig. 6(b) the evaluation result using the trained CNN is shown. First the CNN is applied to all raw spectrograms and then the spectral centroid algorithm is applied to the denoised output of the CNN in the same manner as in Fig. 6(a). The result shows that most of the features are preserved by the CNN but some filtering of the data is apparent, especially at low range between 10:00 o'clock and noon. However, the CNN outputs a stable wind speed in the low SNR regime beyond the ABL up to the clouds at a range between 4000 m and 6000 m. The CNN is therefore able to significantly extend the maximum usable range of the measured data. For further processing, a combination of the results from Fig. 6(a) and Fig. 6(b) can be used. This is demonstrated in

Optics EXPRESS

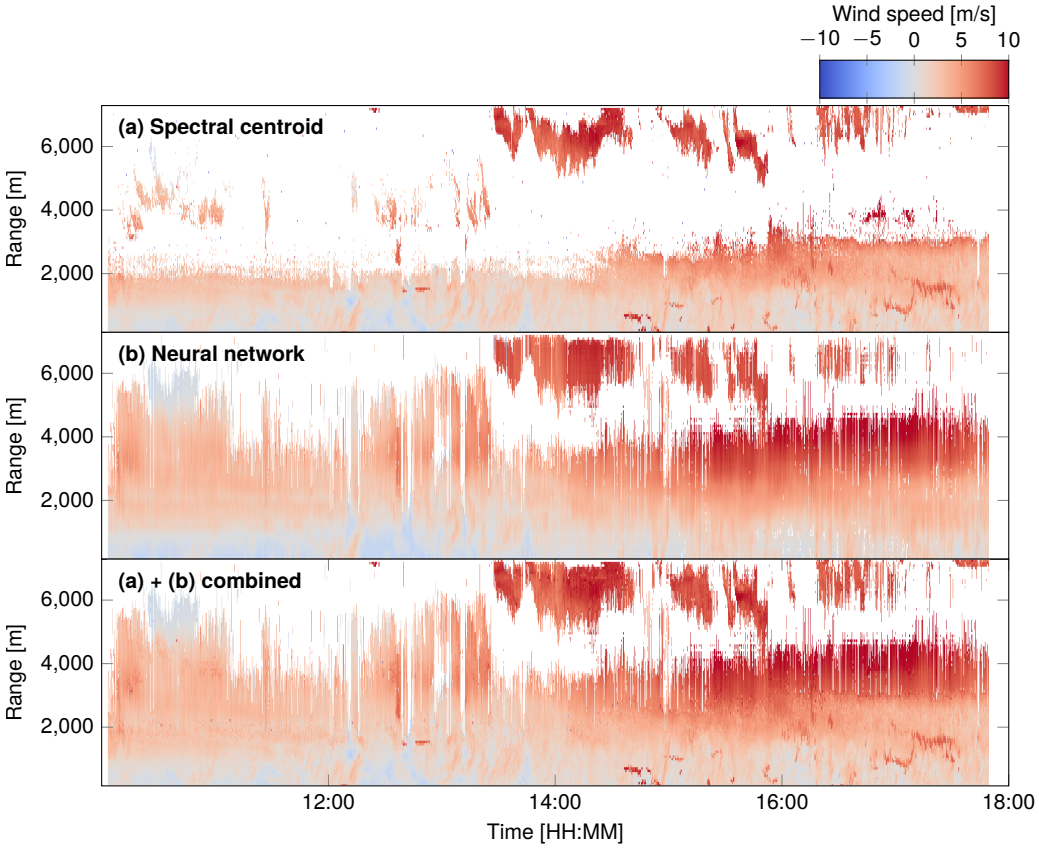


Fig. 6. Line-of-sight wind speed plots for a measurement on July 27th, 2021 for the comparison of spectral centroid and neural network evaluation. The time axis shows the local time (GMT+2). A detailed description of the visible features and comparison of the algorithms is given in the text. (a) Spectral centroid algorithm for frequency estimation. Note that the structures that appear in the near-ground region after 15:00 are false-positive detections by the spectral centroid algorithm due to range gate aliasing. (b) Output of the neural network with spectral centroid post-processing. (c) Combination of the data from spectral centroid and neural network into a single plot.

Fig. 6(c). This plot uses the data from the direct spectral centroid algorithm wherever possible but fills any gaps with data using the CNN. It can be seen that only few combination artifacts are visible at the edge of the ABL, indicating that the CNN output is consistent with the direct spectral centroid evaluation. The aliasing artifacts are effectively filtered by the CNN. This is especially remarkable as the aliasing artifacts are not filtered when generating labeled training data.

For a quantitative comparison the number of evaluable range gates, i.e. ranges gates with a signal above the threshold signal to noise value, is compared between both evaluation algorithms. This measure is not free of systematic biases (e.g. low-SNR artifacts or the dependency on the signal structure) and does not take into account the true accuracy of the frequency estimators. However, this comparison serves as a reasonable example of the achievable improvement of our suggested denoising method. In Fig. 7 the normalized histogram of the ratio of evaluable range gates for the data in Fig. 6 is shown. The ratio is expressed as percentage, where 0 % means

Optics EXPRESS

that both algorithms lead to the same number of evaluable range gates and for 100 % twice the number of range gates could be used for evaluation. The result shows that a drop in performance by using the neural network is observed in very few cases. The peak increase in performance is 40 % to 50 %. Once again it should be stressed that this result is dependent on the measurement conditions. For a clear air measurement with low aerosol optical depth the advantage of using our method is higher than on days with high aerosol optical depth and low altitude clouds, where the spectral centroid estimator is sufficient.

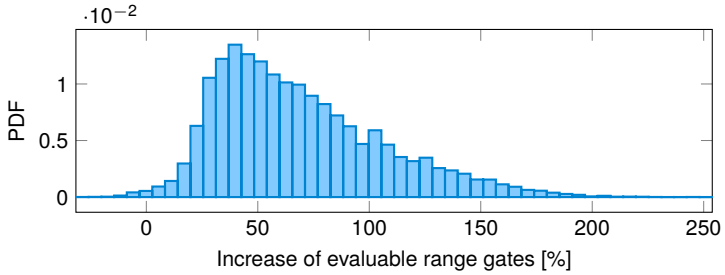


Fig. 7. Normalized histogram of the increase of usable range gates by using the neural network for evaluation based on the data shown in Fig. 6. The increase is the ratio between the number of evaluable range gates in Fig. 6(b) to Fig. 6(a). The y-axis shows the probability density function (PDF).

As a final test and further validation of the CNN denoising method, we have compared the resulting measured wind speed of the lidar with the wind velocity as measured by radiosondes launched by the German Meteorological Service (Deutscher Wetterdienst, DWD). At the DWD location in Stuttgart (48.8281 °N, 9.2006 °E) radiosondes are launched three times a day. As our measurement site is located about 11.3 km away (48.7490 °N, 9.1024 °E) a comparison is possible under certain conditions. If the wind conditions are almost constant over the ascension time and the wind field is constant in the whole region, the projected velocity onto the lidar's measurement direction is comparable. Unfortunately this is not often the case due to the topography around Stuttgart.

However, a lidar measurement performed on September 14$^{\text{th}}$, 2021 with an elevation angle of 45° has been performed where the conditions above have been fulfilled. The measurement result is shown in Fig. 8. Again the spectral centroid (a) and the neural network evaluation (b) are shown. The radiosonde launched around 10:45 UTC. The ascension time from 321 m altitude (above sea-level) to 3000 m was 442 s. Taking all lidar LOS-speeds during this period of time for all range gates results in the value range as plotted by the filled area in Fig. 8(c). The radiosonde data contains only the GPS position and altitude. From this data, the trajectory $\boldsymbol{r}_{\text{RS}}$ is calculated. The velocity $\boldsymbol{v}_{\text{RS}} = d\boldsymbol{r}_{\text{RS}}/dt$ is given by the first derivative of the trajectory. It is assumed that no up-/downwind components are present by setting the z-axis velocity to zero $\hat{\boldsymbol{v}}_{\text{RS}} = (v_x, v_y, 0)$. With this assumption the z-velocity component of the radiosonde originates from buoyancy only. The projection onto the lidar line-of-sight with unit vector $\boldsymbol{e}_{\text{Lidar}}$ is given by $\hat{\boldsymbol{v}}_{\text{RS}} \cdot \boldsymbol{e}_{\text{Lidar}}$. The result is shown as solid red line in the figure, where a moving average with a window of 10 has been applied to smooth the data. The observed comparison shows a reasonable match between the radiosonde data and the speed measured by the wind lidar for the neural network evaluation approach. As expected, the distribution of measured wind speeds increases with altitude but the projected radiosonde velocity lies within the uncertainty range. The mismatch at low altitudes can be explained by the radiosonde's trajectory, which showed significant trundling up to 800 m altitude. The measurement also shows a good agreement between the spectral centroid and the CNN evaluation. In the future, a measurement with the wind lidar and a radiosonde launched

Optics EXPRESS

at the same location could provide further validation of the presented neural network low SNR signal recovery.

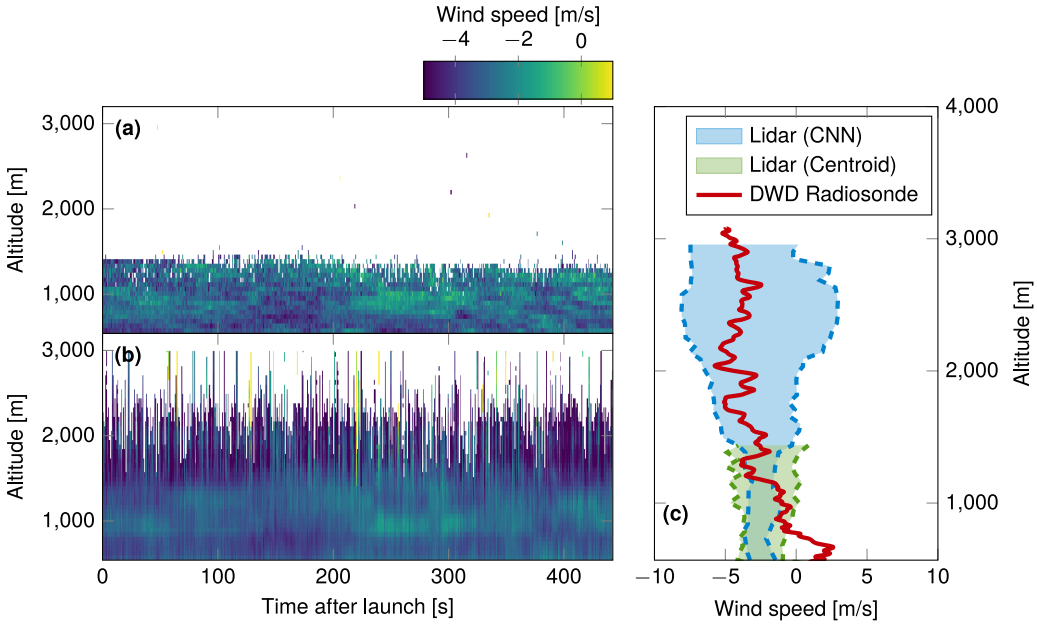


Fig. 8. Short section of a lidar measurement on September 14th, 2021 at 10:45 UTC which coincides with the launch time of a DWD radiosonde. The altitude is referred to above sea-level. (a) Wind speed as extracted from raw data using the spectral centroid algorithm. (b) Neural network data processed by spectral centroid algorithm during the same time period. (c) Comparison of the projected radiosonde velocity onto the lidar line-of-sight with the wind speed as extracted from (a) and (b) during the time where the radiosonde ascended from the launch site to an altitude of about 3000 m. The lidar velocity is depicted as the filled area between the dashed curves for the two different algorithms. All lidar data points within this time period fall within the marked area.

6. Conclusion

An all-fiber coherent Doppler wind lidar operating at 1550 nm with FPGA real-time data pre-processing and a convolutional neural network as new approach for denoising the wind spectrograms have been presented. The neural network is trained using a combination of labeled measured spectrograms and simulated lidar spectrograms. Using this denoising method the maximum usable lidar range can be extended beyond the atmospheric boundary layer and extract information from low SNR data.

The feasibility of this method has been demonstrated for a typical measurement with a high variability in the signal structure. For this particular measurement a 40 % to 50 % increase in the number of evaluable range gates occurred with the highest incidence. In general, the benefit of applying our proposed method is highly dependent on the observed meteorological conditions.

A first validation using radiosonde data is presented. Future measurements at the radiosonde launch site could lead to improved validation data. Additionally, a different optimized convolutional neural network architecture could be used to improve the robustness of the network. Long-short-term-memory (LSTM) recurrent neural networks could be used to include the temporal evolution of the lidar spectrogram into the neural network's abstraction ability.

Optics EXPRESS

Acknowledgments. We would like to thank the German Meteorological Service for the operation of the radiosonde service and publicly providing the trajectory data. We thank Nico Miller for experimental support, Luis Mandl for his helpful comments and Samantha Siegert for proof-reading the manuscript.

Disclosures. The authors declare no conflicts of interest.

Data availability. Data underlying the results presented in this paper are not publicly available at this time but may be obtained from the authors upon reasonable request.

Supplemental document. See Supplement 1 for supporting content.

References

1. O. Reitebuch, Wind lidar for atmospheric research, in Atmospheric Physics: Background – Methods – Trends, U. Schumann, ed. (Springer, 2012), pp. 487–507.
2. Y. Käsler, S. Rahm, R. Simmet, and M. Kühn, "Wake measurements of a multi-mw wind turbine with coherent long-range pulsed doppler wind lidar," J. Atmospheric Ocean. Technol. **27**(9), 1529–1532 (2010).
3. A. C. P. O. Nijhuis, L. P. Thobois, F. Barbaresco, S. D. Haan, A. Dolfi-Bouteyre, D. Kovalev, O. A. Krasnov, D. Vanhoenacker-Janvier, R. Wilson, and A. G. Yarovoy, "Wind hazard and turbulence monitoring at airports with lidar, radar, and mode-s downlinks: The ufo project," Bull. Am. Meteorol. Soc. **99**(11), 2275–2293 (2018).
4. R. Targ, B. C. Steakley, J. G. Hawley, L. L. Ames, P. Forney, D. Swanson, R. Stone, R. G. Otto, V. Zarifis, P. Brockman, R. S. Calloway, S. H. Klein, and P. A. Robinson, "Coherent lidar airborne wind sensor ii: flight-test results at 2 and 10 μm," Appl. Opt. **35**(36), 7117–7127 (1996).
5. B. Witschas, S. Rahm, A. Dörnbrack, J. Wagner, and M. Rapp, "Airborne wind lidar measurements of vertical and horizontal winds for the investigation of orographically induced gravity waves," J. Atmospheric Ocean. Technol. **34**(6), 1371–1386 (2017).
6. T. Kanitz, J. Lochard, J. Marshall, P. McGoldrick, O. Lecrenier, P. Bravetti, O. Reitebuch, M. Rennie, D. Wernham, and A. Elfving, Aeolus first light: first glimpse, in *International Conference on Space Optics – ICSO 2018*, vol. 11180 Z. Sodnik, N. Karafolas, and B. Cugny, eds., International Society for Optics and Photonics (SPIE, 2019), pp. 659–664.
7. R. M. Hardesty, "Performance of a discrete spectral peak frequency estimator for doppler wind velocity measurements," IEEE Trans. Geosci. Remote Sensing **GE-24**(5), 777–783 (1986).
8. B. Lottman and R. Frehlich, "Evaluation of doppler radar velocity estimators," Radio Sci. **32**(2), 677–686 (1997).
9. P. Du, W. A. Kibbe, and S. M. Lin, "Improved peak detection in mass spectrum by incorporating continuous wavelet transform-based pattern matching," Bioinformatics **22**(17), 2059–2065 (2006).
10. C. Yang, Z. He, and W. Yu, "Comparison of public peak detection algorithms for MALDI mass spectrometry data analysis," BMC Bioinf. **10**(1), 4–13 (2009).
11. A. Sherstobitov, V. Banakh, A. Nadeev, I. Razenkov, I. Smalikho, and A. Falits, "Development of an all-fiber coherent doppler lidar in the IAO SB RAS," EPJ Web Conf. **237**, 06005 (2020).
12. S. Abdelazim, D. Santoro, M. Arend, F. Moshary, and S. Ahmed, Development of all-fiber coherent doppler lidar system for wind sensing, Tech. rep., American Meteorological Society (2010).
13. O. Kliebisch and P. Mahnke, "Real-time laser doppler anemometry for optical air data applications in low aerosol environments," Rev. Sci. Instrum. **91**(9), 095106 (2020).
14. G. Bi and E. Jones, "A pipelined fft processor for word-sequential data," IEEE Trans. Acoust., Speech, Signal Process. **37**(12), 1982–1985 (1989).
15. M. A. Mohandes and S. Rehman, "Wind speed extrapolation using machine learning methods and lidar measurements," IEEE Access **6**, 77634–77642 (2018).
16. A. García-Gutiérrez, D. Domínguez, D. López, and J. Gonzalo, "Atmospheric boundary layer wind profile estimation using neural networks applied to lidar measurements," Sensors **21**(11), 3659 (2021).
17. S. Abdelazim, D. Santoro, M. Arend, F. Moshary, and S. Ahmed, Signal to noise ratio characterization of coherent doppler lidar backscattered signals, in *EPJ Web of Conferences*, vol. 119 (EDP Sciences, 2016), p. 17014.
18. Y. A. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, Efficient backprop, in Neural Networks: Tricks of the Trade: Second Edition, G. Montavon, G. B. Orr, and K.-R. Müller, eds. (Springer, 2012), pp. 9–48.
19. G. Peeters, "A large set of audio features for sound description (similarity and classification) in the cuidado project," CUIDADO Ist Project Report **54**, 1–25 (2004).
20. R. Engelmann, >Aerosol vertical exchange in the convective planetary boundary layer, Ph.D. thesis, Faculty of Physics and Earth Sciences, University of Leipzig (2009).
21. F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <0.5mb model size, Arxiv, 1602.07360 (2016).
22. J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah, "Julia: A fresh approach to numerical computing," SIAM Rev. **59**(1), 65–98 (2017).
23. M. Innes, E. Saba, K. Fischer, D. Gandhi, M. C. Rudilosso, N. M. Joy, T. Karmali, A. Pal, and V. Shah, Fashionable modelling with flux, CoRR **abs/1811.01457** (2018).
24. S. W. Henderson, D. R. Philip Gatt, and R. M. Huffaker, Wind lidar, in Laser Remote Sensing, T. F. Takashi Fujii, ed. (CRC Press, 2005), chap. Wind Lidar, pp. 469–723.
25. D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, (2017).