Assessing super-resolution GANs for randomly-seeded particle fields

A. Güemes[1], C. Sanmiguel Vila[2], S. Discetti[1,*]

1: Aerospace Engineering Research Group, Universidad Carlos III de Madrid, Spain
2: Sub-Directorate General of Terrestrial Systems, Spanish National Institute for Aerospace Technology (INTA), Ctra. M-301, Km 10,500, 28330, San Martín de la Vega, Spain
*Corresponding author: sdiscett@ing.uc3m.es

Keywords: PIV processing, Machine Learning, PTV, super-resolution.

ABSTRACT

In this work, we demonstrate and assess the performances of a novel super-resolution generative adversarial network (GAN) framework. The algorithm, recently introduced by the authors (Güemes et al., 2022), leverages random spatial sampling in particle images to provide incomplete views of the high-resolution underlying fields. The main novelty is that the architecture, named Randomly Seeded GAN (RaSeedGAN), does not need full high-resolution training samples. The training is performed directly using the sparse sensors (e.g. particles) available in each snapshot, reduced on a regular grid by spatial averaging in bins. Bins without vectors are simply skipped during training. Provided that the particles randomly sample the space within the dataset, it is possible to reconstruct the mapping from low to high-resolution input with such incomplete "gappy" views. The proposed technique is tested on several synthetic datasets based on simulations, ocean surface temperature distribution measurements, and particle image velocimetry data of a zero-pressure-gradient turbulent boundary layer flow. While the applications in this work target fluid mechanics examples, the proposed method can be applied to more general frameworks where mapping is performed by moving sensors and/or with random on-off status. For further details and a more extensive comparison against the state of the art, the reader is referred to the pre-print of the full article available at `https://arxiv.org/abs/2202.11701`.

1. Introduction

In laser-based optical velocimetry, the flow velocity (and temperature in laser-based thermometry) is normally sampled at specific locations by particles, and a flow field is inferred on a regular grid. Several approaches have been proposed for this purpose, being the most popular one based on cross-correlation multi-pass iterative algorithms. Particle Image Velocimetry (PIV) is substantially founded on this paradigm, although with many variants. Nonetheless, this process can also be substantially formulated as a reconstruction problem from sparse sensors. Various approaches can be followed to obtain full-field measurements from incomplete representations depending on the sparsity level and the number of the available fields. Reconstruction techniques based on extracting flow features with Proper Orthogonal Decomposition (POD Berkooz et al., 1993) have been

broadly used. A remarkable example is the Gappy-POD filling procedures (Everson & Sirovich, 1995; Venturi & Karniadakis, 2004; Raben et al., 2012), based on estimating non-gappy POD modes and time coefficients by least-squares regression on the available sparse PTV data. Recently **?** proposed a technique merging gappy POD and ensemble-particle-averaging techniques. The method, referred to as Data-Enhanced Particle Tracking Velocimetry (DEPTV), is based on building ensemble POD modes by linear stochastic estimation on temporal modes of low-resolution measurements. DEPTV does not require a high-resolution dictionary and builds directly enhanced measurements using low-resolution spatially-averaged fields and sparse fields. The main drawback is the required large number of samples to achieve convergence if the POD eigenspectrum is not sufficiently compact. Tirelli et al. (2022) proposed another approach referred to as K-Nearest Neighbour-PTV (KNN-PTV) that relies on merging information from PTV vectors of different non-time-resolved snapshots enforcing local similarity. The methodology is based on first splitting the data domain into subdomains and then generating a feature space in each of them using a local POD.

For cases where the number of available snapshots is small, compressed sensing is also an interesting pathway based on reconstructing fields using a sparse representation on a universal basis (Huang, 2013; Manohar et al., 2018; Callaham et al., 2019). The main issue of these methods is that they rely on a linear approximation. This approximation results in less robust reconstructions in the presence of strong non-linearities and for highly corrupted measurements, thus limiting their applicability to relatively low gappyness levels.

Recently, neural network-based approaches have been largely exploited as a tool to build flexible non-linear relations between fluid flow data (Güemes et al., 2019; Maulik et al., 2020; Gao et al., 2021; Erichson et al., 2020; Sun & Wang, 2020; Fukami et al., 2021; Arzani et al., 2021; Gundersen et al., 2021). In the field of super-resolution imaging, architectures based on generative adversarial networks (GANs, Goodfellow et al., 2014) have been successfully applied also in turbulent flows (Fukami et al., 2019; Z. Deng et al., 2019; Kim et al., 2021). The paradigm onto which these techniques are based is on training a deep neural network to generate realistic high-resolution fields from low-resolution measurements. The main limitation of these methods is that they require a library of examples to train the reconstruction algorithm, i.e. a set of low and high-resolution measurements (although not necessarily labelled).

Recently, we proposed a novel super-resolution GAN approach to estimate field quantities without needing full-resolution fields for training (Güemes et al., 2022). This is a significant advantage, as virtually all particle-based measurements can be enhanced with this method, with no special setting for the experimental setup. The algorithm leverages the random sampling of the particles in space to train super-resolution GANs with only partial views of the high-resolution underlying fields. This technique has been referred by the authors as Randomly-Seeded super-resolution GAN (RaSeedGAN, Güemes et al., 2022). The main advantage of the presented methodology is that it relies on the information of sensors that are not required to be in a fixed location and can also be in a random on-off status, such as tracers in particle tracking velocimetry or thermogra-

phy. Pairs of low-resolution snapshots created by a simple binning procedure and high-resolution sparse snapshots are used for the training process, making it unnecessary to have a complete high-resolution field as the target for the neural network training process and preventing the overfitting of the model.

This work includes a thorough assessment of the technique based on synthetic and experimental test cases. The methodology is briefly presented in Section 2. A set of challenging validation cases is included in Section 3, with a performance assessment based on reconstruction accuracy and physical consistency. Finally, the conclusions are drawn.

2. Methodology

The proposed RaSeedGAN algorithm is based on the working principle of GANs (Goodfellow et al., 2014). The GANs architecture comprises two competing agents (most often modelled by deep neural networks): a generator, which produces artificial output mimicking reality from a statistical input distribution; a discriminator, which distinguishes genuine samples from generated ones. In this work, we adopt the super-resolution architecture proposed by Ledig et al. (2017), with the minor modification of the removal of the batch-normalization layers, following the findings of Wang et al. (2018). The architecture is sketched in Figure 1.

The generator of RaSeedGAN receives complete full-field information in input, whose resolution has to be enhanced (thus, from now on, we refer to it as "low-resolution", L_R). Such input is produced, for instance, by interpolation of scattered data, Gaussian process regression, or Voronoi tessellation Fukami et al. (2021). For particle images, the generator's input could be composed of the velocity fields from cross-correlation-based analysis or by interpolation of scattered vectors. In this work, we adopt a simple spatial-averaging binning procedure. The bin size is selected to have, on average, approximately 10 sensors per bin. This condition aligns with the classical rule-of-thumb of Particle Image Velocimetry (PIV). To each bin, a value is assigned as the averaged quantity measured by the sensors.

The discriminator network is fed with both real and generated fields. The main novelty is that the real fields are sparse (we refer to it as "high-resolution", H_R), i.e. they are obtained by binning the scattered sensor/vector distributions assigning 0 to the empty bin, and the average of the measures of the sensors falling within the bin otherwise. The empty bins are skipped during the training process. As output, the discriminator provides a probability of the input being real (0s) or generated (1s).

The generator loss is computed as the weighted mean-squared error of the output regarding its target, the latter obtained with the binning procedure described above. Target with smaller bin sizes will clearly have a higher percentage of bins without sensors/vectors, i.e. an increased sparsity level. The loss function is thus computed only considering bins containing vectors.

During training, also an adversarial loss is included. This loss is defined as the binary cross-

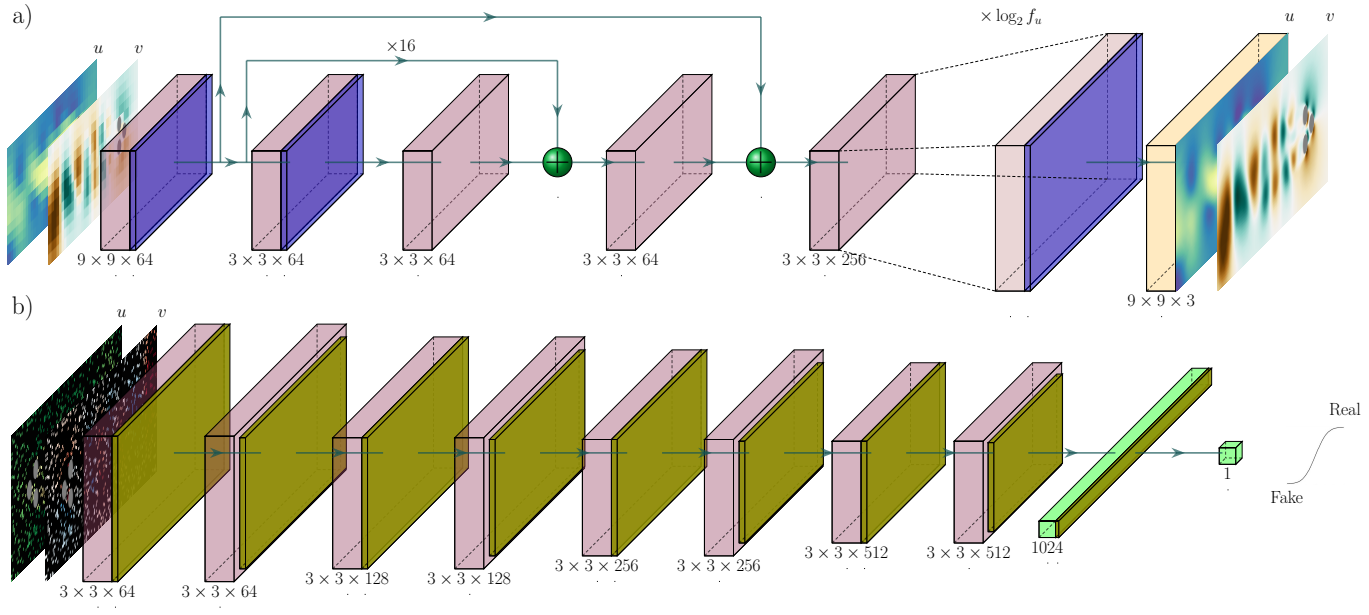

Figure 1. Schematic illustration of GAN architecture. a) Generator; b) Discriminator. The novel element is the use for training and testing of gappy high-resolution fields provided directly by binned particle tracking data.

entropy of the predicted fields. The discriminator loss is thus defined as the mean values between the binary cross-entropy of the predicted fields and the target ones.

The discriminator loss function is defined as

$$\mathcal{L}_D = -\mathbb{E}[\log D(H_R)] - \mathbb{E}[\log(1 - D(F_V \otimes G(L_R)))] \tag{1}$$

Here $D(\cdot)$ indicates the output of the discriminator and represents the probability that the field is real or fake, and $\mathbb{E}[\cdot]$ represents the operation of taking the average in the training mini-batches. Clearly, the discriminator is trained to assign 1 when fed by the original high-resolution fields H_R. On the other hand, it should also be trained to assign 0 to the generated fields $G(\cdot)$, obtained by feeding the generator with the low-resolution fields L_R and masked with F_V to account only for locations that are not empty in the binned target. The mask F_V contains 0 in locations where the bin in the target is empty and 1 where it carries at least one sensor/vector. The loss function of the generator network is defined as:

$$\mathcal{L}_G = \sum_{i=1}^{Nx} \sum_{j=1}^{Nz} |H_{R_{i,j}} - F_V \otimes G(L_R)_{i,j}|^2 + \lambda \mathcal{L}_D \tag{2}$$

where λ is a scalar parameter set to 0.001 to weight the contribution of the adversarial loss and N_X, N_Z are the number of grid points in the two dimensions of the image.

The network architectures are implemented in the open source framework Tensorflow (Abadi et al., 2016).

3. Results

The proposed algorithm is assessed on various synthetic and experimental test cases. Two synthetic test cases with different challenges are considered: velocity data of the wake of a cluster of three cylinders, which has a clear shedding signature; velocity data of a turbulent channel flow at moderate Reynolds number, characterized by a larger wealth of scale. Furthermore, the algorithm performance is proven in two experimental test cases: temperature data from the NOAA sea surface temperature database and PIV data of a zero-pressure-gradient (ZPG) turbulent boundary layer (TBL).

3.1. Synthetic test cases

3.1.1. Description of the datasets

Test Case 1 is based on a Direct Numerical Simulation (DNS) of a 2D wake flow around a cluster off three equally-spaced cylinders, referred to as *fluidic pinball* (N. Deng et al., 2020). The cylinders, with radius $R = D/2$, form an equilateral triangle with a side length equal to $3R$. The triangle points upstream, with the downstream side orthogonal to the freestream flow. The axis of the cylinders of the downstream side are located at $x = 0$, with the $y-$axis positioned at the mean point of the two cylinder axis. The Reynolds number is set equal to $Re_D = 130$ (referred as chaotic regime N. Deng et al., 2020). More details of the simulation and flow behaviour are reported in N. Deng et al. (2020).

Synthetic PTV data are generated from the DNS dataset. In total, 4737 fields are generated with randomly located vectors. The dataset is split into training (4000 fields) and testing (the remaining 737 fields). A domain covering $x/D \in [-5, 8.04]$, $y/D \in [-3.84, 3.84]$ is discretized with 25pix$/D$. This results in 576×192 pixel discretization. For each snapshot 1100 scattered vectors are generated, corresponding to an image density of 0.01 vectors per pixel.

The low-resolution images for training resemble the typical PIV output by binning regions to 32×32 pixels with 50% overlap. This corresponds to approximately 10 vectors per each bin. Although this is a farily simple modelling of PIV processing, this can be considered as the simplest approximation of it based on solely the same scattered vectors that will be later used for the training of RaSeedGAN. In Test Case 1, we aim to achieve upsampling factors up to 8. For an upsampling factor f_u, the bin size is set $b = D_I/f_u$, with D_I being the bin/window size of the low-resolution data. The overlap between bins is maintained in the upsampling process, i.e. the grid is refined by the same amount. Reducing the bin size increases the percentage of empty bins (from here on referred as "gappyness"). The gappyness level for Test Case 1 is $11\%, 57\%, 87\%$ for $f_u = 2, 4, 8$, respectively.

Test Case 2 is generated from a DNS dataset of a turbulent channel flow available at the Johns Hopkins Turbulence Database (`http://turbulence.pha.jhu.edu/`). The full domain of the

simulation has a size of 2 half-channel-heights h from wall to wall, $8\pi h$ in the streamwise direction and $3\pi h$ in the spanwise direction. The DNS database covers one channel-flow through time $8\pi h/U_b$ (where U_b is the channel bulk velocity) with a DNS time step of $\delta t = 0.0014 h/U_b$. For more details on the simulation, the reader is referred to Li et al. (2008). Our synthetic test case is based on subdomains of $2h \times h$ in the streamwise and wall-normal directions, respectively, discretized with $512\text{pix}/h$. As in the previous case, an image density of 0.01 vectors per pixel is set. Independent snapshots are obtained by considering a time separation of 1 convective through time in each subdomain and extraction of subdomains in different regions in the streamwise and spanwise directions (similarly to Raiola et al., 2015). In total, 11856 snapshots have been generated. The training database comprises 10000 snapshots, while the remaining ones are used for testing. The low-resolution dataset is built mimicking a PIV averaging process with a bin size of 32×32 pixels, 50% overlap. As in the previous case, f_u up to 8 is sought. The gappyness level is $12\%, 43\%, 85\%$ for $f_u = 2, 4, 8$, respectively.

It must be remarked that, in both cases, the binned distributions are generated using the original positions directly on the particles, i.e. particle images are not generated. The effect of noise is introduced with a controlled random perturbation on velocity vectors of realistic intensity. In Test Case 1, we have introduced Gaussian noise with a standard deviation equal to 0.01 pixels, i.e. 1% of the freestream displacement. For Test Case 2, we included a noise of 0.1 pixels to account for the more significant freestream displacement if compared to the previous test case. Such error levels can be considered realistically representative of the expected errors in Particle Tracking Velocimetry data with reasonable image quality. This approach attempts to isolate the errors due to spatial resolution limitations without introducing uncontrolled uncertainty due to feature matching and outliers in the process. Such aspects are instead addressed by the experimental test cases.

3.1.2. Performance analysis

The performances of RaSeedGAN are first qualitatively assessed with the instantaneous representation of reconstructed fields. An instantaneous realisation for the synthetic test cases is reported in Figure 2. The panel includes: the low-resolution (LR) field used in the input, representative of the original PIV data; the binned data for $f_u = 8$ used for training, referred to as "Sparse HR (high-resolution) Reference" since it includes 0 for empty bins; the reconstruction by RaSeedGAN at $f_u = 8$; the complete high-resolution data, include only for a quality check, although not used at any moment during the training.

In Test Case 1, the RaSeedGAN recovers the structure of the near-wake correctly behind the cylinders. This fact is particularly evident in the crosswise velocity component v, where the attenuation of the intensity of the fluctuations in the near field is significantly reduced, and peaks of velocity nearby the cylinders are recovered. The description of the momentum deficit in the streamwise velocity component u is similarly enhanced with respect to the low-resolution field.

For Test Case 2, similar qualitative considerations can be drawn. The recovery of small scales

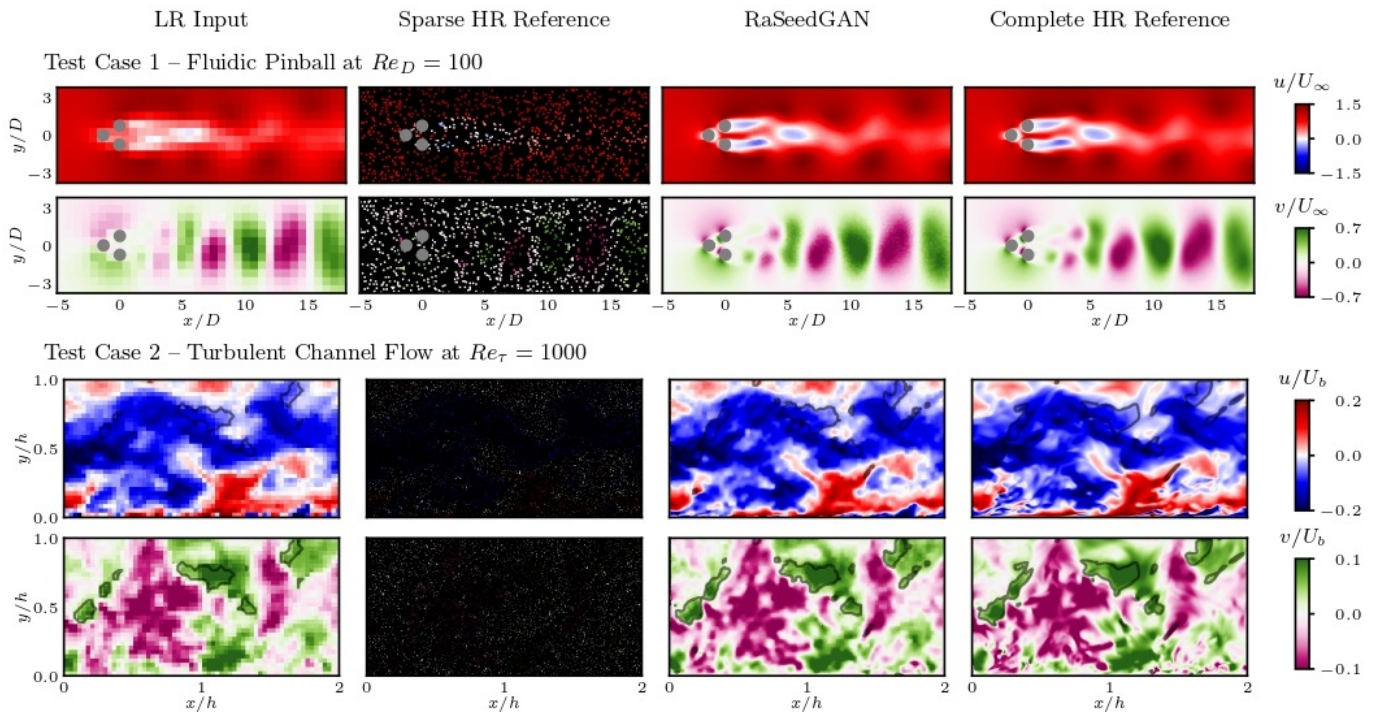

Figure 2. Instantaneous flow field realizations for Test Case 1 and 2. From left to right, the columns indicate the low-resolution (LR) input, the sparse high-resolution (HR) binned distributions (black regions indicate empty bins), the reconstruction obtained with RaSeedGan with $f_u = 8$ and the complete DNS field included for reference. First and second row for each test case indicate respectively streamwise (u) and cross-wise (v) velocity components. Black contour lines in Test Case 2 identify *uvsters*. Velocities are presented in normalized form using the freestream velocity U_∞ for Test Case 1 and the bulk velocity U_b for Test Case 2.

and wall-attached vertical velocity fluctuations in the near-wall region is noticeable. In order to better highlight the improvement of the flow field resolution, we have included contours of *uvsters* (Lozano-Durán et al., 2012), i.e. extreme Reynolds-shear-stress events. The *uvsters* description seems vastly improved by RaSeedGAN if compared to the original low-resolution input.

Table 1 includes the pixel-based root mean squared error of the two synthetic test cases for different upsampling levels. The error of a standard cubic interpolation on a regular grid of the velocity vectors is included in the parenthesis. The standard deviation of each quantity is used to normalise the corresponding error. For Test Case 1, a significant improvement with respect to cubic interpolation is observed. For Test Case 2, the difference between the two methods is smaller, possibly due to the richer wealth of scales of the channel flow if compared to the wake of the fluidic pinball.

3.2. Experimental test cases

3.2.1. Description of the datasets

Test Case 3 is based on the NOAA sea surface temperature (`http://www.esrl.noaa.gov/psd/`). The dataset contains temperature data collected by satellites and ships. We have con-

Table 1. Pixel-based root mean squared error for the synthetic test cases. The results are scaled with the standard deviation of fluctuating quantities. Errors of cubic interpolation are included in parenthesis.

	Test Case 1		Test Case 2	
f_u	u	v	u	v
2	0.304 (0.956)	0.237 (0.511)	0.274 (0.314)	0.312 (0.363)
4	0.207 (1.544)	0.178 (0.720)	0.276 (0.309)	0.302 (0.351)
8	0.163 (2.176)	0.149 (1.004)	0.282 (0.317)	0.312 (0.351)

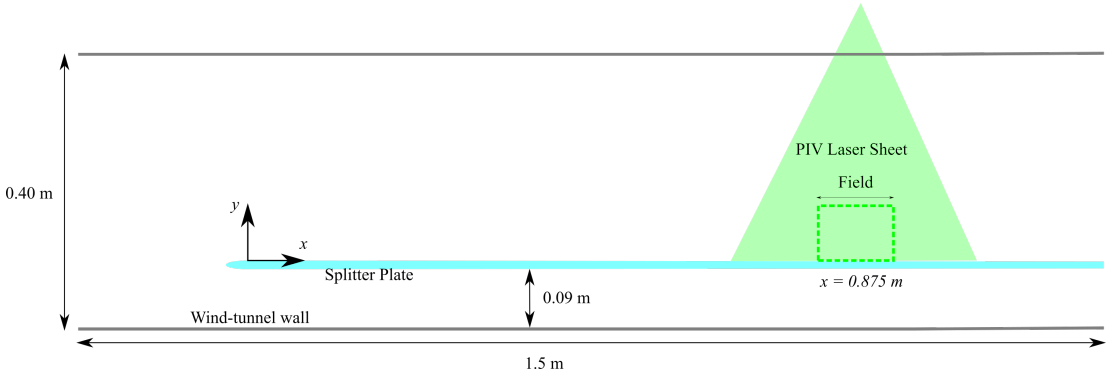

Figure 3. Experimental setup for Test Case 4 (from Guemes et al., 2019). The ZPG TBL develops on the flat plate in light blue in the figure.

sidered 7305 temperature fields, covering the period from January 1^{st} 2000 to December 21^{st} 2019. The data are interpolated on a regular grid composed of 720×1440 points. The training dataset comprises 6000 snapshots, while the remaining ones have been used for testing. We have extracted temperature measurements at 10300 randomly-located positions (different in each snapshot), creating a framework similar to the previous test cases. A bin size of 32×32 points has been set to build the low-resolution input database. This corresponds to 10 points on average per bin. The gappyness level is 36%, 69% and 90% for upsampling factors of $f_u = 2, 4, 8$.

Test Case 4 is based on a wind-tunnel PIV experiment on a zero-pressure gradient turbulent boundary layer. The experimental setup is briefly described here and sketched in Figure 3; for a more detailed description, the reader is referred to Guemes et al. (2019). The flow has been seeded Di-Ethyl-Hexyl-Sebacate (DEHS) droplets with 1 μm diameter, generated by a Laskin nozzle. The seeded flow has been illuminated by a dual cavity Ng:Yag Quantel Evergreen laser (200 mJ/pulse at 10 Hz). An ANDOR Zyla sCMOS 5.5 MP camera (2560 $\times$ 2160 pixel array, 6.5 $\times$ 6.5 μm pixel size) has been used to capture the PIV images. The spatial resolution has been set to about 48400 pix/m. The camera is equipped with a Tokina 100 mm lens. The lens aperture is set to $f_{\#} = 16$, and the objective is set slightly out of focus in order to increase the particle image size and reduce peak locking artificially. Laser reflections and illumination background have been removed in pre-processing using the POD-based approach proposed by Mendez et al. (2017).

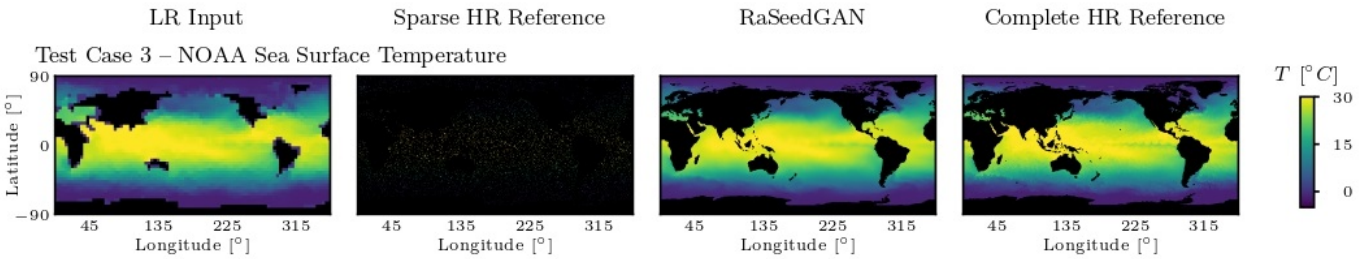

Figure 4. Measurement of ocean surface temperature from NOAA sea surface data. The columns, from left to right, indicate the low-resolution (LR) input, the sparse high-resolution (HR) binned distributions (black regions indicate empty bins), the reconstruction obtained with RaSeedGan with $f_u = 8$ and the complete DNS field included for reference.

The full dataset contains 39000 images. Vectors are obtained using a super-resolution particle tracking Keane et al. (1995) with a standard multi-pass window deformation algorithm (Scarano, 2001) to build a solid predictor. The code used for PIV processing is a custom-made version developed by the University of Naples (Astarita, 2008). The density of vectors is 0.006 vector per pixel, thus resulting in about 25000 vectors in the 2048×2048 pixels image area. For training 30000 snapshots are used, with the remaining ones used for testing.

To test RaSeedGAN with a solid ground truth reference, we fictitiously decrease the vector density by a factor of 10 by arbitrarily deleting vectors. A cross-correlation-based analysis on the original images with 32×32 pixels window is used to generate the high-resolution reference. The low-resolution fields are estimated with an analysis of the original images with 128×128 pixel windows with 50% overlap with the same multi-step image deformation algorithm mentioned earlier Scarano (2001). The 128×128 pixels window contains on average 10 particles on the reduced-density dataset, corresponding to a mean particle spacing of around 40 pixels. The sparse HR distributions used during the training of RaSeedGAN are obtained by binning the reduced-density vector distributions with upsampling factors $f_u = 2, 4$. This corresponds to gappyness levels of 15% and 61%, respectively.

It must be remarked again that, as in the synthetic test cases, the high-resolution fields are only used as a final quality check, and they are not exploited in any form in the training process.

3.2.2. Performance analysis

An instantaneous temperature field visualisation from Test Case 3 of the NOAA sea surface temperature with upsampling factor $f_u = 8$ is presented in Figure 4. The reconstruction is compared with the original dataset interpolated on the 720×1440 points. From a qualitative inspection, the detail of the temperature fields and the reconstructed profile of the continents is a clear indication of enhanced resolution. It must be remarked, though, that the enhanced description of the continent is a natural consequence of the refined grid and the absence of data in such regions. Also, in this test case, the error level reported in Table 2 for cases with upsampling levels of $f_u = 2 - 8$ is

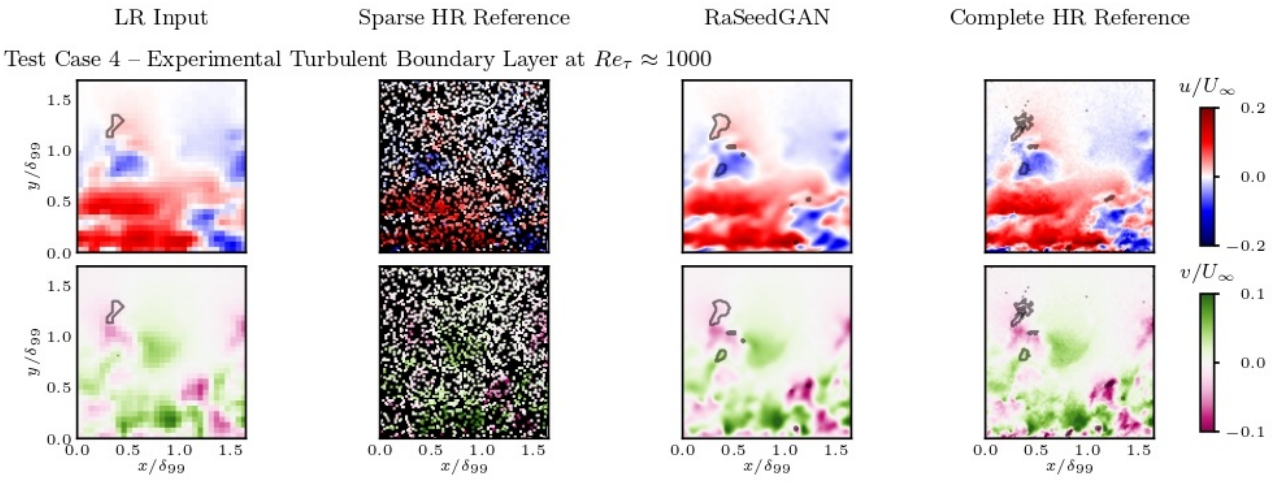

Figure 5. Instantaneous flow field realization for the PIV experiment on a turbulent boundary layer. The columns indicate, from left to rigth, the low-resolution (LR) input with bin of 128×128 pixels, the sparse high-resolution (HR) binned distributions (black regions indicate empty bins), the reconstruction obtained with RaSeedGAN with $f_u = 4$ and the outcome of standard PIV processing with 32×32 pixel window. First and second row for each test case indicate respectively streamwise (u) and cross-wise (v) velocity components. Velocities are presented in normalized form using the freestream velocity U_∞.

Table 2. Pixel-based root mean squared error for the experimental test cases. The results are scaled with the standard deviation of fluctuating quantities. Errors of cubic interpolation are included in parenthesis

	Test Case 3	Test Case 4	
f_u	T	u	v
2	0.718 (0.742)	0.204 (0.738)	0.275 (0.743)
4	0.720 (1.087)	0.177 (0.642)	0.228 (0.642)
8	0.716 (0.780)	-	-

lower than the obtained using standard cubic interpolation of the scattered data.

In Figure 5 the streamwise and wall-normal velocity fields of the zero-pressure-gradient turbulent boundary layer flow are reported. The contours also include *uvsters*, as in the previous section, indicated with black lines. Unlike Test Case 2, noise contamination is not simulated nor imposed here, as it directly arises from feature matching uncertainty.

Similarly to Test Case 2, we observe a recovery of wall-attached vertical velocity fluctuations in the near-wall region. However, the reconstruction quality is lower than the one achieved in Test Case 2. This result was expected since the quality of the experimental data is lower than the synthetic data. Furthermore, the ground truth for comparison is also based on experimental data, with their inherent uncertainty. As evidence of this statement, the difference between RaSeedGAN and the standard cubic interpolation is more remarkable than in Test Case 2, as reported in Table 2.

We include a comparison of turbulent flow statistics for further performance assessment. It must be remarked that RaSeedGAN is not trained to recover flow statistics but rather to reconstruct field

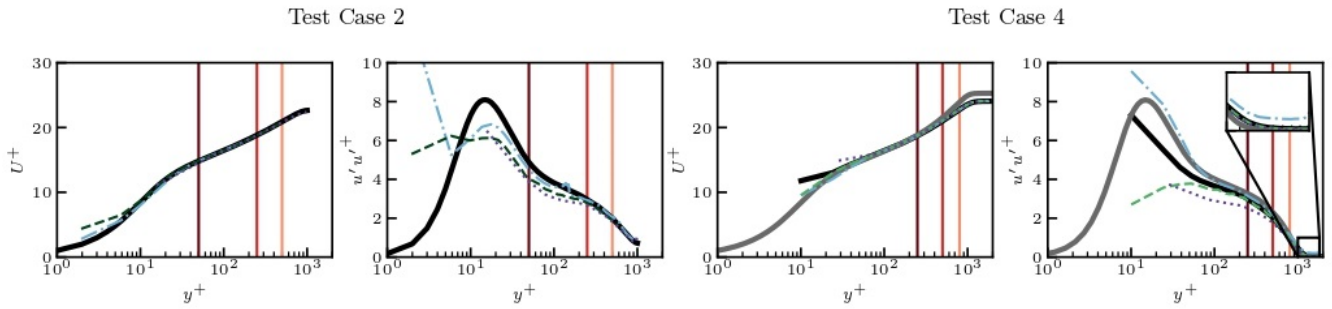

Figure 6. Inner-scaled mean streamwise velocity and variance profiles for Test Case 2 (DNS turbulent channel flow) and Test Case 4 (experimental turbulent boundary layer). Lines represent the data used to calculate the quantity: high-resolution reference (solid black), original low-resolution measurements (dotted purple), cubic interpolation (cyan dashed-dotted) and RaSeedGAN reconstruction (dashed green, with $f_u = 8$ for Test Case 2 and $f_u = 4$ for Test Case 4). The inset in the variance profile of Test Case 4 provides a magnified view of the region with a wall distance above 1000 inner units. Vertical green lines (from dark to light for increasing distance from the wall) indicate stations where the streamwise turbulent spectra are computed. For Test Case 4, grey profiles refer to data from (Schlatter & Örlü, 2010).

features; consequently, this assessment can be considered a fair metric for faithful flow reconstruction. The wall-normal profiles of first and second-order statistics of the streamwise velocity are reported in Figure 6. Test Case 2 is also included in this section for the completeness of the information. An inner scaling is used (with inner-scaled quantities indicated with the superscript $+$), i.e. the statistics are scaled with the friction velocity u_τ, and the wall-normal coordinate y is scaled with the viscous length scale $\ell^* = \nu/u_\tau$. An ensemble averaging over the testing dataset and spatial averaging along the streamwise direction is performed. For Test Case 2, the reference profiles are directly available from the Johns Hopkins Turbulence Database. For Test Case 4, the reference profile is extracted from the high-resolution PIV analysis on the original dataset. For completeness, in Test Case 4, we also include profiles from friction-Reynolds-number matched simulations Schlatter & Örlü (2010).

RaSeedGAN provides a solid improvement of the turbulence statistics profiles compared to the low-resolution training data. Cubic interpolation, at first glance, seems to have even superior performances, especially below 15 wall units (corresponding to about 8 pixels in Test Case 2 and 16 pixels in Test Case 4). This result might be explained due to a tendency of RaSeedGAN to inherit intensity modulation error from the low-resolution input, while instead, cubic interpolation is directly performed on the pointwise data. On the other hand, the instantaneous fields obtained from cubic interpolation have higher noise contamination. This fact might induce a bias in the variance estimation, which can compensate for modulation issues. This misleading offset effect has been documented by Atkinson et al. (2014) in the general framework of PIV/PTV measurements. Observing the velocity profiles at a large wall-normal distance, we observe indeed that the plateau of the variance is higher for cubic interpolation, thus hinting towards higher noise contamination (inset of Figure 6).

Streamwise velocity spectra are thus computed at a different wall-normal distance to address this

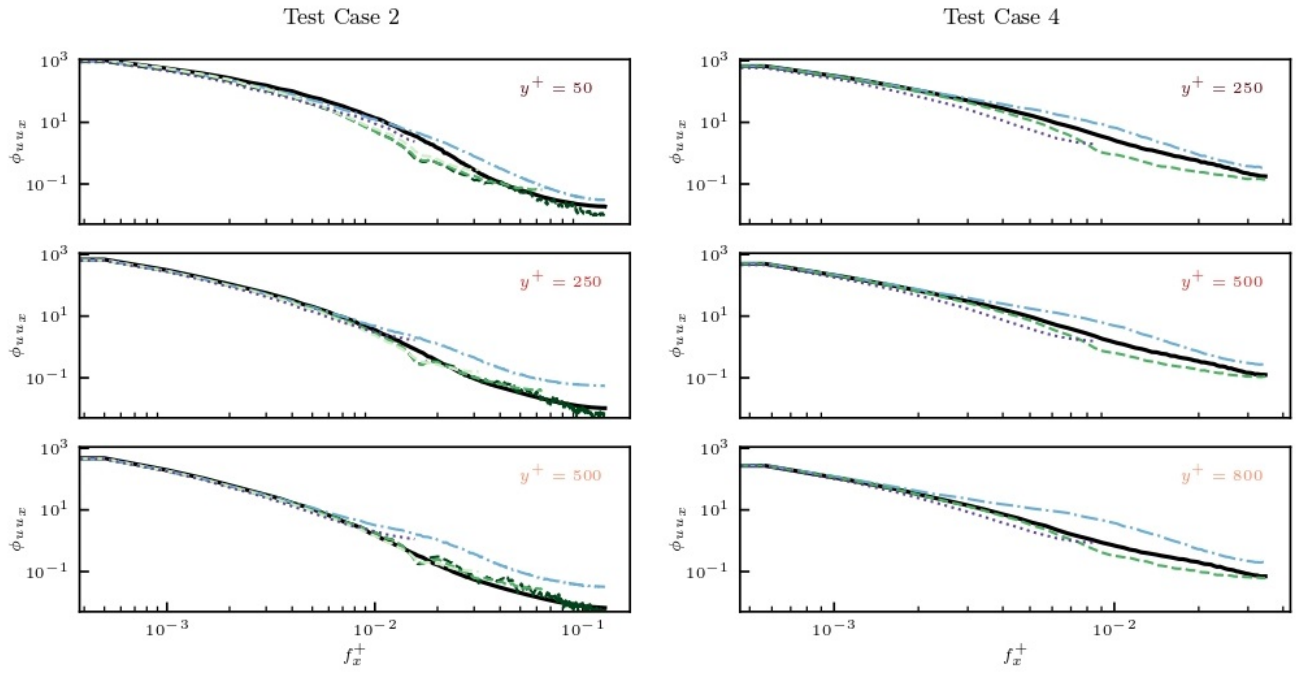

Figure 7. Streamwise turbulent spectra at different wall distances. Left: Test Case 2. Right: Test Case 4. Lines represents the data used to calculate the quantity: high-resolution reference (solid black), low-resolution original measurements (dotted purple), cubic interpolation (cyan dashed-dotted) and RaSeedGAN reconstruction (dashed green, light to dark going from $f_u = 2$ to 8).

effect in terms of the spectral content of the flow scales. A comparison is included in Figure 7. Except for small wall distances, RaSeedGAN significantly increases the reach of the spectrum. The spectrum obtained from cubic interpolation peels quickly away from the reference already at a relatively large scale. This fact hints at more significant white-noise contamination. The cut-off frequency, estimated as the frequency at which the spectrum of the low-resolution data and RaSeedGAN reach 80% of the intensity of the reference spectrum, is increased by at least a factor of 3 except for $y^+ = 50$ of Test Case 2.

4. Conclusions

A novel deep-learning approach based on generative adversarial networks to perform super-resolution reconstruction of sparse measurements has been assessed. The proposed method exploits the ability of the neural networks to learn mapping functions even when some layers have randomly ignored neurons during the training process. The proposed RaSeedGAN obtains a non-linear mapping to high-resolution reconstructed images, using as inputs the low-resolution images obtained from the sparse measurements by a bin-averaging process and as targets the original sparse measurements. To prove the validity of the methodology in real-life applications, different examples that include numerical and experimental data from velocity and temperature measurements were successfully tested, demonstrating the accuracy and robustness of the proposed

method. For further details and a more extensive comparison against the state of art the reader is referred to the pre-print of the full article available at `https://arxiv.org/abs/2202.11701`.

Acknowledgements

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 949085). NOAA High Resolution SST data provided by the NOAA/OAR/ESRL PSL, Boulder, Colorado, USA. The authors warmly acknowledge N. Deng, B. Noack, M. Morzynski and L. Pastur for providing the code for the fluidic pinball simulations.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., ... others (2016). Tensorflow: A system for large-scale machine learning. In *12th USENIX symposium on operating systems design and implementation (OSDI 16)* (pp. 265–283).

Arzani, A., Wang, J.-X., & D'Souza, R. M. (2021). Uncovering near–wall blood flow from sparse data with physics–informed neural networks. *Physics of Fluids*, *33*, 071905.

Astarita, T. (2008). Analysis of velocity interpolation schemes for image deformation methods in PIV. *Experiments in Fluids*, *45*(2), 257–266.

Atkinson, C., Buchmann, N. A., Amili, O., & Soria, J. (2014). On the appropriate filtering of piv measurements of turbulent shear flows. *Experiments in fluids*, *55*(1), 1–15.

Berkooz, G., Holmes, P., & Lumley, J. L. (1993). The proper orthogonal decomposition in the analysis of turbulent flows. *Annu. Rev. Fluid Mech.*, *25*(1), 539–575.

Callaham, J. L., Maeda, K., & Brunton, S. L. (2019). Robust flow reconstruction from limited measurements via sparse representation. *Physical Review Fluids*, *4*(10), 103907.

Deng, N., Noack, B. R., Morzyński, M., & Pastur, L. R. (2020). Low-order model for successive bifurcations of the fluidic pinball. *Journal of Fluid Mechanics*, *884*, A37. doi: 10.1017/jfm.2019.959

Deng, Z., He, C., Liu, Y., & Kim, K. C. (2019). Super-resolution reconstruction of turbulent velocity fields using a generative adversarial network-based artificial intelligence framework. *Physics of Fluids*, *31*(12), 125111.

Erichson, N. B., Mathelin, L., Yao, Z., Brunton, S. L., Mahoney, M. W., & Kutz, J. N. (2020). Shallow neural networks for fluid flow reconstruction with limited sensors. *Proc. Roy. Soc. A*, *476*(2238), 20200097.

Everson, R., & Sirovich, L. (1995). Karhunen–loeve procedure for gappy data. *JOSA A*, *12*(8), 1657–1664.

Fukami, K., Fukagata, K., & Taira, K. (2019). Super–resolution reconstruction of turbulent flows with machine learning. *Journal of Fluid Mechanics*, *870*, 106–120.

Fukami, K., Maulik, R., Ramachandra, N., Fukagata, K., & Taira, K. (2021). Global field reconstruction from sparse sensors with voronoi tessellation-assisted deep learning. *Nature Machine Intelligence*, *3*, 945—951.

Gao, H., Sun, L., & Wang, J.-X. (2021). Super-resolution and denoising of fluid flow using physics-informed convolutional neural networks without high-resolution labels. *Physics of Fluids*, *33*(7), 073603.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., … Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, *27*.

Güemes, A., Discetti, S., & Ianiro, A. (2019). Sensing the turbulent large-scale motions with their wall signature. *Physics of Fluids*, *31*(12), 125112.

Guemes, A., Ianiro, A., & Discetti, S. (2019). Experimental assessment of large-scale motions in turbulent boundary layers. In *13th international symposium on particle image velocimetry.*

Güemes, A., Vila, C. S., & Discetti, S. (2022). Super-resolution gans of randomly-seeded fields. *arXiv preprint arXiv:2202.11701*.

Gundersen, K., Oleynik, A., Blaser, N., & Alendal, G. (2021). Semi-conditional variational auto-encoder for flow reconstruction and uncertainty quantification from limited observations. *Physics of Fluids*, *33*(1), 017119.

Huang, X. (2013). Compressive sensing and reconstruction in measurements with an aerospace application. *AIAA J.*, *51*(4), 1011–1016.

Keane, R., Adrian, R., & Zhang, Y. (1995). Super-resolution particle imaging velocimetry. *Measurement Science and Technology*, *6*(6), 754.

Kim, H., Kim, J., Won, S., & Lee, C. (2021). Unsupervised deep learning for super-resolution reconstruction of turbulence. *Journal of Fluid Mechanics*, *910*.

Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., … others (2017). Photo-realistic single image super-resolution using a generative adversarial network. In *Proc. ieee cvpr* (pp. 4681–4690).

Li, Y., Perlman, E., Wan, M., Yang, Y., Meneveau, C., Burns, R., … Eyink, G. (2008). A public turbulence database cluster and applications to study lagrangian evolution of velocity increments in turbulence. *Journal of Turbulence*(9), N31.

Lozano-Durán, A., Flores, O., & Jiménez, J. (2012). The three-dimensional structure of momentum transfer in turbulent channels. *Journal of Fluid Mechanics*, *694*, 100–130.

Manohar, K., Brunton, B. W., Kutz, J. N., & Brunton, S. L. (2018). Data-driven sparse sensor placement for reconstruction: Demonstrating the benefits of exploiting known patterns. *IEEE Contr. Syst. Mag.*, *38*(3), 63–86.

Maulik, R., Fukami, K., Ramachandra, N., Fukagata, K., & Taira, K. (2020). Probabilistic neural networks for fluid flow surrogate modeling and data recovery. *Physical Review Fluids*, *5*, 104401.

Mendez, M., Raiola, M., Masullo, A., Discetti, S., Ianiro, A., Theunissen, R., & Buchlin, J.-M. (2017). POD-based background removal for particle image velocimetry. *Exp. Thermal Fluid Sci.*, *80*, 181–192.

Raben, S. G., Charonko, J. J., & Vlachos, P. P. (2012). Adaptive gappy proper orthogonal decomposition for particle image velocimetry data reconstruction. *Measurement Science and Technology*, *23*(2), 025303.

Raiola, M., Discetti, S., & Ianiro, A. (2015). On piv random error minimization with optimal pod-based low-order reconstruction. *Experiments in Fluids*, *56*(4), 1–15.

Scarano, F. (2001). Iterative image deformation methods in PIV. *Measurement Science and Technology*, *13*(1), R1.

Schlatter, P., & Örlü, R. (2010). Assessment of direct numerical simulation data of turbulent boundary layers. *Journal of Fluid Mechanics*, *659*, 116–126.

Sun, L., & Wang, J.-X. (2020). Physics-constrained bayesian neural network for fluid flow reconstruction with sparse and noisy data. *Theor. Appl. Mech. Lett.*, *10*(3), 161–169.

Tirelli, I., Ianiro, A., & Discetti, S. (2022). An end-to-end knn-based piv approach for high-resolution measurements and uncertainty quantification. *arXiv preprint arXiv:2205.02766*.

Venturi, D., & Karniadakis, G. E. (2004). Gappy data and reconstruction procedures for flow past a cylinder. *Journal of Fluid Mechanics*, *519*, 315–336.

Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., … Loy, C. C. (2018). ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks. In *Proc. eur. conf. comput. vis. workshop (2018)* (pp. 63–79).