



Research papers

Flow measurement in open channels using imaging techniques in conjunction with a convolutional neural network

Hao-Che Ho^{*}, Yu-Wei Chiu, Ting-Yu Chen, Yen-Cheng Lin

Department of Civil Engineering, National Taiwan University, Taipei, Taiwan

ARTICLE INFO

This manuscript was handled by A. Bardossy, Editor-in-Chief, with the assistance of Fi-John Chang, Associate Editor

Keywords:

Flow measurement
LSPIV
Direct cross-correlation
Convolutional neural network
Image technique

ABSTRACT

Data related to river velocity and discharge are important for water resource management. Non-intrusive image measurement techniques based on direct cross-correlation (DCC) algorithms, such as particle image velocimetry (PIV), are widely used to measure velocity and discharge in the field. However, environmental noise is highly complex and uncontrollable in the field, significantly reducing the accuracy of DCC-based methods. Convolutional neural networks (CNNs) are commonly used in image recognition because of their outstanding accuracy, which far exceeds that of conventional imaging methods. However, these accuracy levels cannot be directly extrapolated for flow movement estimation. Therefore, in this study, we developed an innovative sub-pixel correction technique that allows CNN-based methods to obtain stable measurements in the PIV framework. This is the first study that successfully applied the concept of CNN to measure velocity data using images. The Hamel-Oseen vortex-flow, uniform steady-flow, and plane laminar jet-flow models are established as benchmark vector fields. Non-uniform illumination and Gaussian noise with varying degrees of interference are added to the synthetic data to evaluate the performance of the CNN-based method in PIV. For noiseless images, the DCC-based and CNN-based methods achieve lower measurement errors than benchmark errors. For noisy images, the DCC suffers a fold error increase ranging from 2.77 to 31.13, whereas the CNN suffers only a fold error increase ranging from 1.25 to 1.68. A 30-m-long flume is then used in an uncontrolled environment to mimic real-world flow measurement. The dispersion of the instantaneous velocity measurements for the CNN is more concentrated than that for the DCC. The acoustic Doppler velocimetry yields an error of only 7.87% in discharge estimation using CNN. These results indicate that the CNN-based method is more robust than conventional methods and has the potential to be effectively applied to measurements in the field.

1. Introduction

In-situ water data are important for water resource management, such as urban planning, flood control construction, river restoration, and flood monitoring. Field data must be obtained to analyze the discharge for flood monitoring, particularly considering the recent increase in the frequency of extreme rainfall events (Lipper et al., 2017; Kouadri et al., 2021). Unfortunately, conventional approaches for the measurement of hydrological phenomena have been hindered by insufficient data resolution, inability to transmit data in real time, and inability to obtain reliable data during storms. Imaging methods have been developed for flow measurement, and their efficacy has been demonstrated in flume experiments (Adrian, 1991; Hann and Greated, 1997; Creutin et al., 2002; Young et al., 2015; Kim et al., 2016). A popular non-intrusive imaging method is the large-scale particle image

velocimetry (LSPIV), which derives flow rates from flow velocity measurements at the surface of river channels (Fujita and Komura, 1994; Fujita et al., 1998; Bradley et al., 2002; Huet et al., 2008; Muste et al., 2014; Ostad-Ali-Askari and Shayan, 2021). This method involves setting up a camera on a riverbank to capture a continuous stream of images of the river surface and derive hydrological data. Image matching is then performed using direct cross-correlation (DCC). Cross-correlation matrices can be calculated directly in the spatial or frequency domain using the fast Fourier transform (FFT). The fact that the DCC algorithm is used to correlate images in space means that the algorithm imposes no limitations on the size of the sub-images (interrogation area, IA) to be compared. Nonetheless, the size of the sub-images can have a significant impact on the computation time and accuracy of the corresponding estimates. The FFT provides notable benefits in terms of computational overhead and suppression of noise related to the inter-correlation

^{*} Corresponding author.E-mail address: haocheho@ntu.edu.tw (H.-C. Ho).

coefficients. FFT also increases the density of the velocity field through multiple channels (Willert and Gharib, 1991; Stamhuis, 2006; Thielicke and Stamhuis, 2014).

The matching accuracy of the DCC method is significantly influenced by the quality of successive images. In indoor flume experiments, it is relatively easy to obtain reliable velocity measurements because the experimental conditions can be easily controlled and high-quality images can be captured. However, environmental conditions (e.g., light, wind, and shadows) cannot be controlled in the field. Despite extensive efforts to eliminate these effects (Chaves, 2012; Zhang et al., 2013; Xu et al., 2017; Zhen et al., 2017; Ghashghaei et al., 2022), most acquired images contain noise that cannot be entirely removed. It is difficult to obtain uniform seeding conditions for particle tracking. The inability to control environmental conditions inevitably undermines the accuracy of the data and limits the applicability of imaging techniques for flow measurement. There is a pressing need for an algorithm that considers not only brightness but also geometric and abstract features. There is also a need for image-processing algorithms that do not rely on a stable environment.

Artificial intelligence (AI) involves using computers to perform tasks that are beyond human cognitive capability. In 1958, Rosenblatt (1958) published a perceptron machine-learning algorithm that was a linear binary classification model. The linearity of the perceptron algorithm was a significant drawback that caused the development of machine learning to be hindered until the 1980s, which was when the multi-layer perceptron algorithm enabled the processing of multiple messages within a network (Makino et al., 1983; Hinton, 1986; Watrous and Shastri, 1987; Morgan and Bourlard, 1990). In recent years, AI and machine learning have been revived with the advent of convolutional neural networks (CNNs), starting with the LeNet-5 network in 1995 (LeCun et al., 1998). CNNs use images as a direct input, thereby eliminating the need for pre-processing image features. CNNs also allow the users to correct the weights of convolutional filters for automatically producing convolutional kernels that enable image recognition. Neural network weights are combined with kernels to reduce the number and complexity of the parameters, while enhancing the tolerance for image distortion, rotation, and panning. CNNs have been widely used in transportation, agriculture, and medicine (Kagaya et al., 2014; Hao et al., 2018; Shang et al., 2018; Traore et al., 2018; Özyurt, 2020; Ma et al., 2021; Ostad-Ali-Askari and Shayannejad, 2021; Xiong et al., 2021). The CNN concept was also implemented to predict discharge and velocity based on empirical and numerical models (Chiang et al., 2017; Forghani et al., 2021; Deng et al., 2022; Karpatne et al., 2022). However, there is no application for in-situ measurements of discharge and velocity with non-intrusive imaging methods. In this study, we employed a CNN as an alternative to the DCC algorithm for the recognition of granular images representing a flow field.

LSPIV is a non-intrusive approach for hydrologic flow measurement; however, it is highly susceptible to environmental conditions. CNNs have proven to be highly effective in resisting the effects of environmental interference; however, no previous study has applied CNNs in the context of LSPIV. Herein, we explored the possibility of replacing the DCC method with the CNN method in LSPIV to allow direct extraction of abstract or geometric features for image matching. In the simulations, we compared both approaches in terms of their accuracy in estimating the flow velocity for various flow scenarios. We also compared the approaches in a full-scale flume experiment in which the surface velocity fields obtained using the two methods were compared with those obtained using acoustic Doppler velocimetry (ADV), for a given cross-sectional surface.

2. Experimental methods

2.1. Concepts underlying the DCC method

The standard operating procedure for particle image velocimetry

(PIV) is to create a grid point on the image sequence to be analyzed as the central location of the extracted IA. The size of the IA is determined by the density of the particles in the window to ensure that this sub-image contains sufficient feature-related information for comparison. The search area (SA) is then defined such that the algorithm can find the IA of highest similarity at time t within the periphery of the grid points at $t + 1$, where the velocity is calculated with sub-pixel correction. The details of the DCC method used to estimate the velocity can be found in Appendix A.

2.2. Design of the proposed CNN scheme

PIV analysis in the current study was performed using the standard procedure for the definition of regions of interest (ROIs) in images and cutting out sub-images of a specific size with fixed-spaced grid points at time t . In the current study, these sub-images were used as input data for the CNN in accordance with the following definition: $X_{64 \times 64}^{g,t}$ ($g = 1\bar{N}$). Each sub-image was defined as a different object class Y^g ; the output of the network was encoded in the form of one-hot (see Fig. 1) to generate an array representing class Y^g .

The data input to the network must be restricted within a specified range through normalization to avoid slow convergence, prevent gradient explosion due to a large difference in data order, and ensure trainability. We stipulated that the intensity of the pixels in an 8-bit grayscale image must take values between 0 and 255. We also normalized the raw image data to 0 s and 1 s before cutting the sub-images.

In the definition of the network structure for the proposed CNN, the characteristics of the cutout sub-images included only the arrangement and shape of the particles, and the time interval between successive images was short. This ensured that the two images did not diverge excessively. To ensure that the network met the requirements in terms of PIV analysis and computation time, we discarded existing network architectures, such as VGG Net, Google Inception, or ResNet. Instead, we built the self-defined convolutional structure shown in Fig. 2. We examined two convolutional layers (Net1) versus four convolutional layers (Net2) as well as maximum pooling layers and a fully-connected layer connected to the output layer.

The depth of the convolutional layer was used to determine the diversity of the extracted features. Essentially, insufficient depth would lead to the extraction of an insufficient number of features to perform recognition tasks, whereas excessive depth would increase the training time. We eventually opted for a two-layer architecture (Net1) with various depth combinations as follows: [8,16], [16,32], [32,64], [64,128], and [128,256]. The number of neurons in the fully-connected layer was also varied as follows: [256], [512], [1024], [2048], and [4096]. We formulated a total of 25 depth/number combinations in the fully-connected layer and then selected the most robust one based on a comparison with the benchmark. We eventually established a convolutional depth of [16,32] with [4096] neurons in the fully-connected layer for all subsequent analyses.

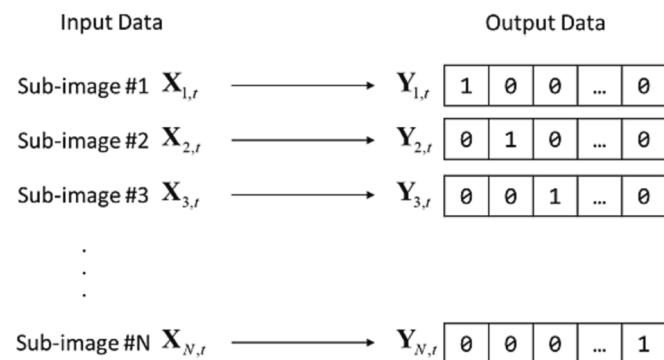


Fig. 1. One-hot encoding method.

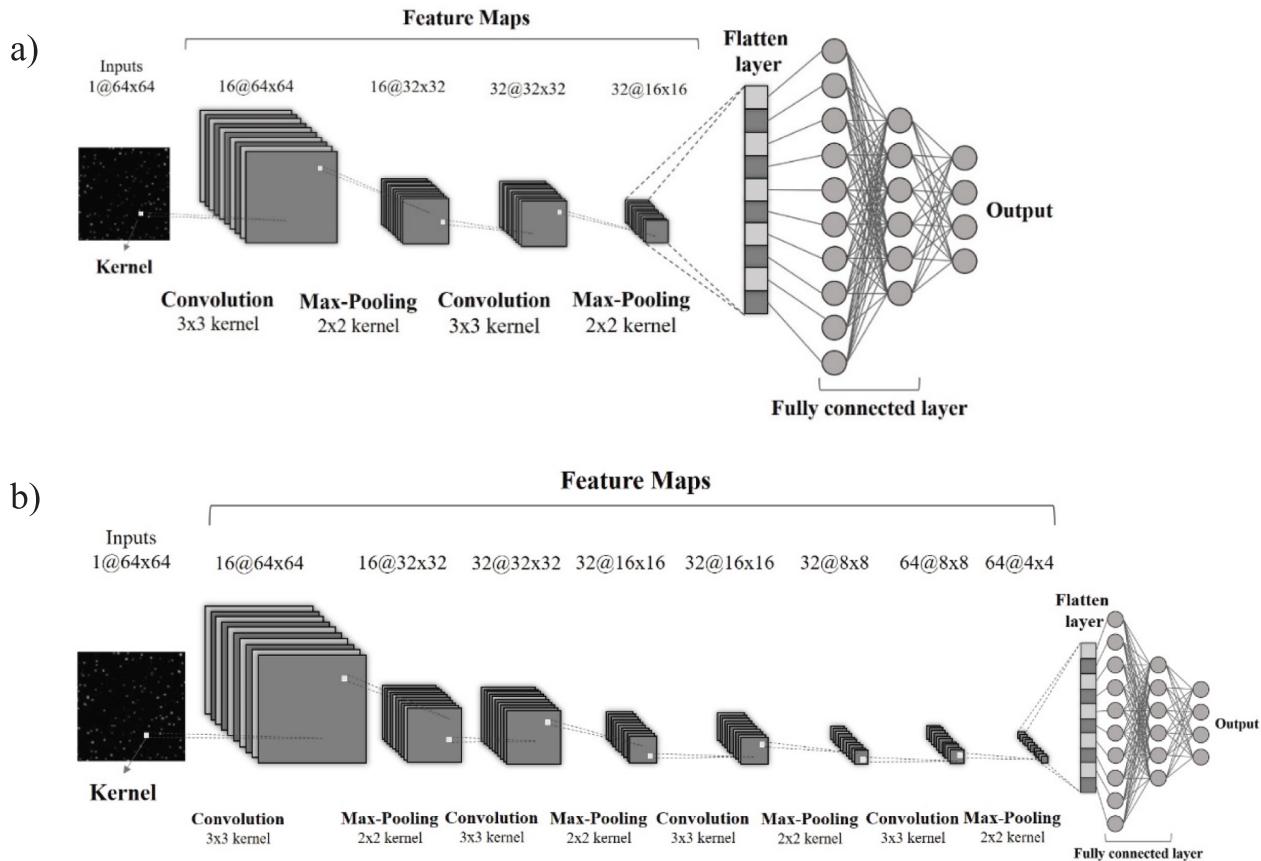


Fig. 2. CNN frameworks proposed for flow measurement: (a) Net1 and (b) Net2.

The depth of the first convolution layer was 16 (Net1), and the depth of the second convolution layer was 32 (Net2). The depths of the third and fourth convolution layers in Net2 were 32 and 64, respectively. The last connection layer included 4096 nodes in the output layer after the fully-connected layer. In the convolution layer, we employed a 3×3 convolutional kernel and a 2×2 pooling kernel in Step 1, which were connected to the fully-connected layer for feature classification. The effects of four activation functions were assessed: the rectified linear unit (ReLU), leaky ReLU (LReLU), parametric rectified linear unit (PReLU), and exponential linear unit (ELU). The details of the four activation functions are provided in Appendix B.

A loss function is used to determine the difference between the output and target values. In the current study, the highly popular cross-entropy (L_i) was used to represent the amount of information between the target array $Y_{1 \times N}$ and network output array $Y'_{1 \times N}$ (De Boer et al., 2005) based on the following equation:

$$L_i = - \sum_{n=1}^N y_{n,i} \log(y'_{n,i}), \quad L_{total} = \sum_i L_i \quad (1)$$

where y is the element of the array Y ; n is the array index and number of classes; N is the total number of classes; and i is the number of arrays. A small cross-entropy value indicates a small difference and vice versa.

Once the loss function is determined, it is necessary to select an optimization algorithm to correct the weights, such that the best solution can be achieved with a minimal discrepancy between the target and network output. The most common methods employed for deep learning or neural networks are the Adam method and stochastic gradient descent (SGD).

In the current study, we opted for the Adam method to avoid the shortcomings of SGD in optimization. The Adam method combines the

momentum algorithm (Qian, 1999) with the Adagrad (Duchi et al., 2011) and RMSProp algorithms. Thus, it corrects errors by adopting a high learning rate at the beginning of training and a lower learning rate in later stages. Moreover, the concept of momentum allows the Adam method to increase or decrease the current error gradient based on the previous error gradient, thereby enabling more rapid correction in the desired direction and slower corrections in other directions.

The Adam method has proven to be effective in most general cases and provides rapid convergence, exceeding that of other optimization algorithms (Kingma and Ba, 2014; Ostad-Ali-Askari et al., 2017). Weight initialization also affects the training and performance of CNNs. Setting a poor initialization can lead to an uneven distribution of data values transmitted in the neural layer, such that the neuron outputs tend to be concentrated around a certain value. A recommended weight initialization suitable for different activation functions was applied in this study (Glorot and Bengio, 2010; He et al., 2015).

To unify the dimensions of the network input data, the sub-images were scaled to the following sizes: 32×32 , 64×64 , 96×96 , and 128×128 . In all convolution layers, the movement step of the convolutional kernel was 1 px, and the initial weights were randomly generated using a normal distribution with a mean value of 0, a standard deviation of 0.1, and an initial bias of 0.01. The softmax function was applied to the output layer. For the proposed CNN model, we also employed batch training with batch regularization to keep the weights in each convolution layer active, thereby stabilizing the performance of the loss function during training and suppressing the network dependence on default values (Ioffe and Szegedy, 2015).

After training was completed using all the sub-images at time t , an SA was defined at the corresponding grid point $(x_{g,t+1}, y_{g,t+1})$ for time $t + 1$ to identify sub-image candidates ($X_{64 \times 64}^{g,k,t+1}$) with K in the SA using the sliding-window approach. All the candidates were input to the network

to obtain the output array $Y_{1 \times N}^{g,k}$ as calculated by the CNN for use in obtaining the loss matrix (L_g). The minimum loss value of the loss matrix was based on the position of the matching sub-image. This made it possible to detect pixel-level movement ($x_{g,t+1}^m, y_{g,t+1}^m$) for use in the calculation of velocity.

2.3. Sub-pixel correction for CNN

The distance represented by a pixel can be scaled for specific applications. For imaging applications using large-scale photographic images of flowing rivers (e.g., LSPIV), the length unit corresponding to a pixel in the image is usually on the centimeter scale. Under these conditions, the measured velocity vector is far from the true value unless the measurement resolution is very high. Sub-pixel correction is an approach for enhancing the measurement resolution.

In conventional PIV, sub-pixel precision corrections are generally obtained via Gaussian curve fitting or 2D Gaussian fitting (Willert and Gharib, 1991; Nobach and Honkanen, 2005). Once the peak (z_{\max}) of the correlation matrix (R_g) is found, the peak of the curve is located through curve fitting using the values at ± 1 pixels adjacent to z_{\max} , which is used to achieve sub-pixel accuracy. However, for the CNN in this study, we replaced the correlation matrix (R_g) with a loss matrix (L_g). The distribution characteristics of the two matrices are not the same. The peak of the loss matrix cannot be fitted using Gaussian curve fitting (see Fig. 3). Theoretically, matching sub-images (X_g^m) should be similar to the original image (X_g), such that if one image were laid over the other, then only minor particle movements would be observed (see Fig. 4). In the current study, we aimed to improve the accuracy at sub-pixel level using the average movement of all particles in the sub-image at microscopic scale as the displacement of sub-pixels. Particle tracking velocimetry (PTV) was used to capture the particle coordinates, and noise in the images was smoothed out using binomial and binomial-Laplacian filters, with the contrast between the bright center and edge of the particle enhanced. As shown in Fig. 5, this smoothed the noise and suppressed the particle edge intensity to enhance edge performance. A quadratic curve was then fitted to the peaks, with a post-filtering intensity exceeding the threshold value (Jähne, 1995; Capart et al., 2002) using the following formula:

$$\begin{cases} x_p = x_i - \frac{1}{2} \frac{I_{i,j+1} - I_{i,j-1}}{I_{i,j+1} - 2I_{i,j} + I_{i,j-1}} \\ y_p = y_i - \frac{1}{2} \frac{I_{i+1,j} - I_{i-1,j}}{I_{i+1,j} - 2I_{i,j} + I_{i-1,j}} \end{cases} \quad (2)$$

where i and j represent the coordinates of the pixel with peak intensity after filtering and I is the intensity of that pixel. The offset between the quadratic maximum and maximum pixel intensity was calculated to obtain the accuracy of the particle coordinates (x_p, y_p) at sub-pixel level.

Thus, by marking the particle coordinates (x_p, y_p) and (x'_p, y'_p) of the matching sub-pixel pairs, the average difference in particle coordinates in the two images can be calculated at microscopic scale, and by adding this movement to the initial results obtained using the CNN, the final movement vector [$\Delta x_g, \Delta y_g$] can be obtained with sub-pixel accuracy.

This approach to particle detection was also applied to determine whether sub-images on grid points included particles or features. For either method (CNN or DCC), if a sub-image were found to contain no particles or distinctive features, that sub-image would not undergo image matching. Regardless of the type of method used, it would be helpful to screen out sub-images without particles or features. Numerous sub-images in the CNN contained evident particles or features; however, some sub-images did not contain particles or evident features because the grid dots fell in areas with an indiscernible particle distribution. Allowing such sub-images to be trained in the network as separate categories would slow down the speed and effectiveness of training. Featureless sub-images would also tend to generate unreliable or erroneous results when using the DCC method. Therefore, we subjected all sub-images to pre-screening.

2.4. Evaluation of measurement results

2.4.1. Root mean square error (RMSE)

The RMSE was used to determine the discrepancy between the estimates of the velocity magnitude and corresponding benchmark values as follows:

$$RMSE = \sqrt{\frac{1}{K} \sum_{k=1}^K (y_k - \hat{y}_k)^2} \quad (3)$$

where y_k represents the velocity vectors calculated using the DCC or CNN methods and \hat{y}_k represents the corresponding benchmark vector at each grid point.

2.4.2. Vector correlation coefficient (VCC)

The velocity vectors estimated using the CNN and DCC methods were compared to the benchmark velocity results using the same grid in order to evaluate the accuracy of both methods based on the VCC. Assuming that the two-dimensional velocity vector obtained using CNN or DCC is $W_1 = u_1 \hat{i} + v_1 \hat{j}$, and that the corresponding benchmark vector is $W_2 = u_2 \hat{i} + v_2 \hat{j}$, then the covariance matrix between both vector fields and the VCC (ρ_v^2) can be calculated. Vector fields W_1 and W_2 are completely uncorrelated if $\rho_v^2 = 0$; however, these vector fields present complete linear correlation if $\rho_v^2 = 2$. When the number of samples in the vector exceeds 64, the distribution of $n\rho_v^2$ approximates a cardinal distribution with four degrees of freedom. This makes it necessary to determine whether the chi-square test values ($X^2 = 9.488$ and $\alpha = 0.05$) for the

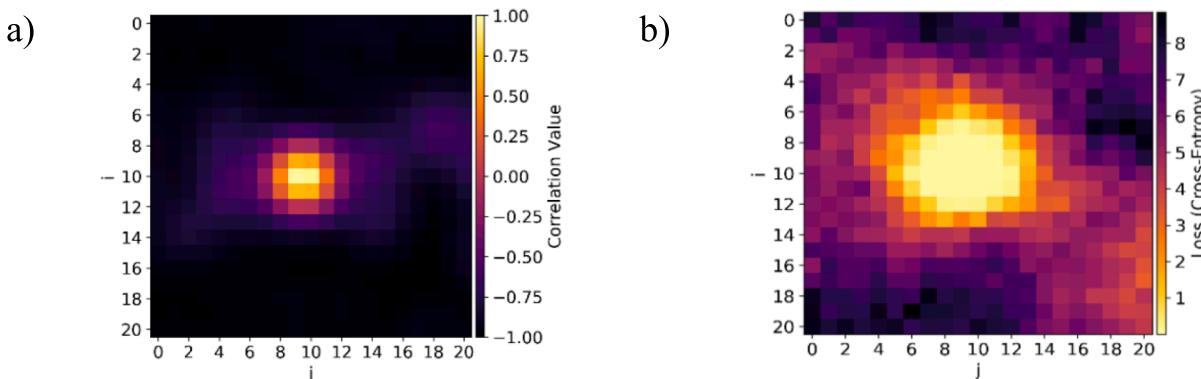


Fig. 3. a) cross-correlation matrix R_g and b) loss matrix L_g .

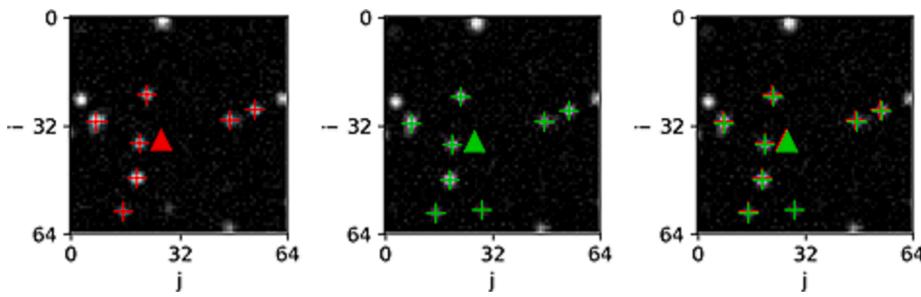


Fig. 4. Particle labeling and overlaying of two sub-images.

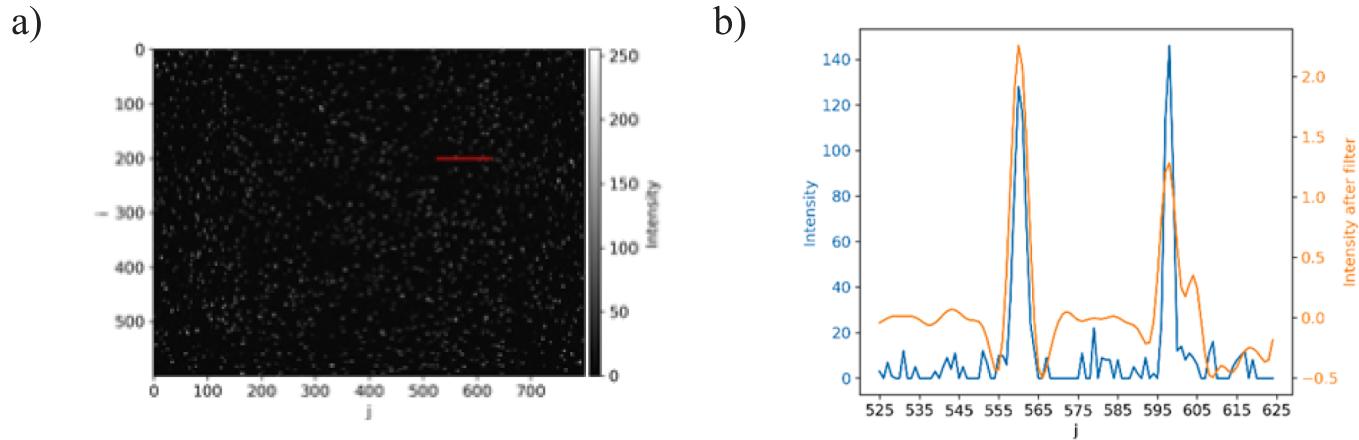


Fig. 5. a) Original image and b) intensity pixel profile at red line in (a).

four degrees of freedom are sufficient. Details of the calculation for the covariance matrix and the VCC (ρ_v^2) can be found in [Appendix A](#).

2.5. Discharge estimation

The goal of using imaging techniques for flow measurement while operating in the field (e.g., LSPIV) implies to obtain an estimate of discharge quickly and easily. The velocity index method, commonly used for discharge estimation, involves multiplying the depth-averaged velocity converted from the surface velocity using an index (α) and the cross-section of the channel. In this study, we conducted a flume experiment to measure discharge and vertical velocity distributions in order to obtain α from ADV measurements. The log-law velocity profile function was used to fit the vertical velocity data for estimating α . The surface flow velocities obtained using the DCC and CNN methods were converted using this index and multiplied by the cross-section to estimate the rate of discharge.

3. Experimental setup

3.1. Benchmark images

The efficacy of the DCC and CNN methods in measuring flow rates was evaluated under various flow fields and environmental conditions using artificial images (as a benchmark) generated using the open-source software program PIVlab based on known true speed values and pixel size (Thielicke and Stamhuis, 2014). The first flow field was set as the Hamel-Oseen vortex (Saffman, 1992) using the following equation:

$$v_\theta(r, t) = \frac{\Gamma}{2\pi r} \left(1 - \exp\left(-\frac{r^2}{r_c^2(t)}\right) \right) \quad (4)$$

$$r_c(t) = \sqrt{4\nu t + r_c^2(0)}, \quad (5)$$

where Γ refers to the circulation, r is the distance from the center of the vortex, ν is the kinematic viscosity, and r_c is the vortex radius.

As shown in [Fig. 6a](#), this describes an up-down, left-right symmetrical twin-turbine velocity field. The use of this velocity field equation made it possible to derive the true velocity vector at each grid point as a benchmark value for comparison. In this study, we set the pixel velocity magnitude to a value two orders of magnitude smaller than 3 and 5 pix/frame.

The second type of flow field was a plane laminar jet flow (see [Fig. 6b](#)). The velocity distribution and flow rate were derived using the following equations (Schlichting, 1933):

$$u = \frac{3}{8\pi} \frac{K}{\nu x} \frac{1}{\left(1 + \frac{\xi^2}{4}\right)^2}, \quad (6)$$

$$v = \frac{1}{4} \sqrt{\frac{3}{\pi}} \frac{\sqrt{K}}{x} \frac{\xi - \frac{\xi^3}{4}}{\left(1 + \frac{\xi^2}{4}\right)^2}, \quad (7)$$

where $K = \frac{J}{\rho}$, $\xi = \sqrt{\frac{3}{16\pi}} \frac{\sqrt{K}}{\nu} \frac{y}{x}$, J is the momentum flux, ρ is the fluid density, and ν is the kinematic viscosity.

The last flow field was a uniform steady flow in which the velocity and flow direction did not change with time or location. In this study, the velocity was -2 pix/frame in the u-direction and $\sqrt{5}$ pix/frame in the v-direction. The flow field is shown in [Fig. 6c](#).

3.2. Artificial image conditions

The artificial images of particles generated using the aforementioned

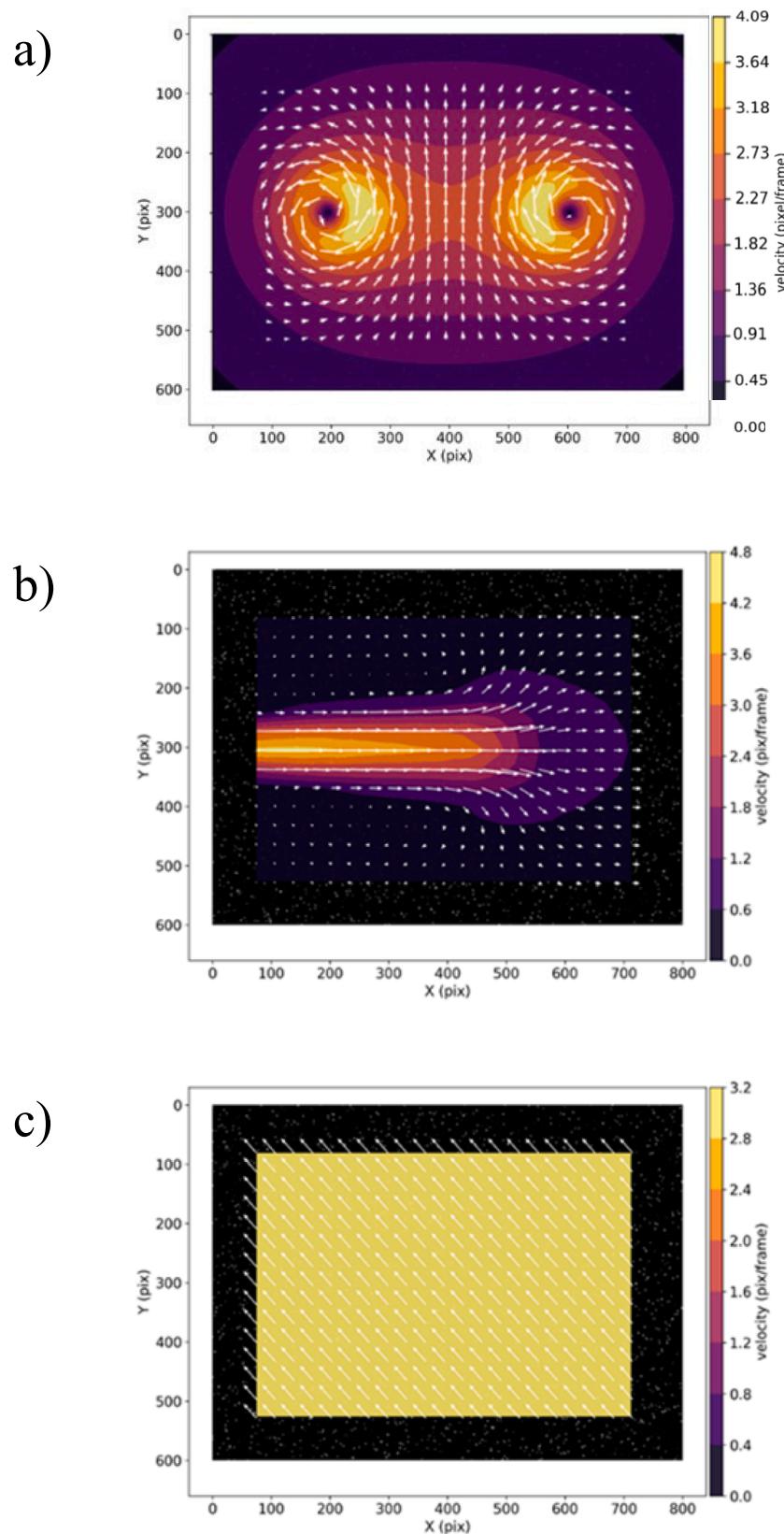


Fig. 6. Vector field superposed on velocity magnitude contours obtained using PIVLab: a) Hamel-Oseen vortex ring; b) plane laminar jet flow; and c) uniform steady flow.

equations and PIVlab presented a high contrast between the particles and a uniform background, which means that the DCC and CNN methods should both be able to obtain precise estimates of particle movement. However, the environmental conditions encountered in the field would likely introduce unexpected errors in flow measurements. Images obtained in the field are subjected to a range of environmental conditions, including uneven brightness and particle density distributions. In this study, we employed particles of two sizes, variations in particle density and particle movement, three light sources, and noisy signals to mimic the interferences typically encountered in the field. The stability and robustness of the DCC and CNN methods were evaluated under these conditions.

3.2.1. Particle distribution effects

The diameter and number of particles were varied to assess the effect of particle density on flow measurement performance. The objective here was to determine whether the CNN method could reduce the negative effects of variations in particle density distribution. Note that even under ideal environmental conditions, images captured in the field are affected by the distance between the camera and water surface as well as the specifications of the recording equipment. Moreover, the pixel area occupied by the particles or features in the image is not fixed. In most field applications, drones are used to capture water surface features. Thus, the resolution of the images is determined by the specifications of the recording equipment and the altitude of the drone above the water (i.e., ground sample distance (GSD)), which can be defined using the following formula:

$$GSD = \frac{H \times s}{f}, \quad (8)$$

where H is the altitude of the drone, s is the pixel size in the sensor, and f is the optical focal length of the lens.

We consider the DJI Phantom 3 drone (4 K version) as an example: its sensor size is 1/2.3', the optical focal length of its lens is 3.57 mm, full high-definition recording is performed at 1080×1920 px, and the image element size is 0.0037 m. The correspondence between altitude H and GSD is presented in Table 1. When measurements are performed on a river presenting a range of 30–50 m in width, the ground resolution decreases inversely with the altitude. For example, at an altitude of 30 m, the GSD would be 31.24 mm/pixel, such that an object measuring 3.1 cm would occupy only one pixel in the image. Note that the leaves are commonly used as a seed for imaging in the range of 5–15 cm². Thus, in these simulations, the diameter of the particles must be 3–5 pixels to ensure that the artificial images accurately represent those obtained in the real world.

In terms of seeding density, the number of particles determines the degree of sparseness in an image. We generated two cases involving 100,000 or 3,000 particles to mimic small-scale indoor experiments and large-scale field experiments, respectively. In the high-density case (100,000 particles), to mimic well-controlled conditions, the particle diameter was set to 3 pixels and the illumination was uniform. In the low-density case (3,000 particles), we aimed to mimic uneven illumination of the water surface by adding noise in the form of an uneven distribution of the background brightness. A total of 22 cases were used to examine the performance of the CNN and DCC methods (see Table 2).

Table 1
Relationship between altitude H and GSD.

H (m)	GSD (mm/pixel)	Equivalent size for image	
		Length (m)	Width (m)
30	31.24	59.984	33.741
35	36.45	69.981	39.364
40	41.66	79.978	44.988
45	46.86	89.975	50.611
50	52.07	99.973	56.235

We then conducted experiments involving two particle velocities (3 or 5 px/frame) in successive images.

Cases V3R1 and V4R1 involved high-density particle distributions simulating small-scale applications, whereas Cases V1R1 and V2R2 involved low-density particle distributions simulating field conditions. In the field, uneven distribution of brightness on the water surface can be largely attributed to cloud cover, shadowing, and water depth. Overcoming this issue generally requires the removal of background noise through tedious image filtering, the results of which are not usually satisfactory. Therefore, we included Cases V1R1N and V2R1N, which added uneven lighting conditions to the conditions in Cases V1R1 and V2R1. Table 2 presents the simulation cases.

3.2.2. Illumination and signal effects

We simulated three illumination conditions representing uniform and non-uniform backgrounds (see Fig. 7). Uniform image illumination involved brightening the background evenly to simulate a lack of reflection in the field. Non-uniform illumination involved adjusting the background brightness to simulate uneven reflections. Note that the same contrast was used for all artificial images. The signal interference caused by the camera sensor and circuitry was replicated by adding Gaussian noise at three densities (see Fig. 8).

3.3. Flume experimental setup

The flume experiment was designed to assess the performance of the CNN and DCC methods based on ground-truth data collected using a SonTek 10-MHz micro ADV. The channel used in this study was 30 m in length and 1 m in width. The camera captured an area of $x = 8.9\text{--}9.6$ m. The cross-section was captured using the ADV measured at $x = 9.27$ m (see Fig. 9a). Ten ground reference points (GRPs) were placed on both sides to facilitate image ortho-rectification. Images were recorded using a 16-megapixel camera (Sony NEX-VG30) at a height of approximately 1.5 m from the bottom of the flume without auxiliary lighting. Video recording was performed using full high-definition resolution at 50 fps with a field of view (FOV) of 1.2×0.7 m². Following ortho-rectification, the image size was reduced to 917×501 px with a resolution of 1.2 mm/pixel. When analyzing the DCC and CNN methods, image pre-processing was not conducted, the IA was fixed at 64 px, and the SA was 12 px extended to the IA (see Fig. 9c). The ADV sampling frequency was set to 25 Hz. The water depth of the flume was 0.249 m. The measurement positions for ADV were at $x = 9.27$ m and $y = 0.15, 0.42, 0.58$, and 0.85 m (see Fig. 9b). For each position, we measured 10 points vertically from the bottom to the water surface using the ADV to capture vertical velocity profiles. For each point, we derived the average velocity from the data obtained over a period of 3 min.

4. Results and discussion

4.1. Evaluation of CNN models

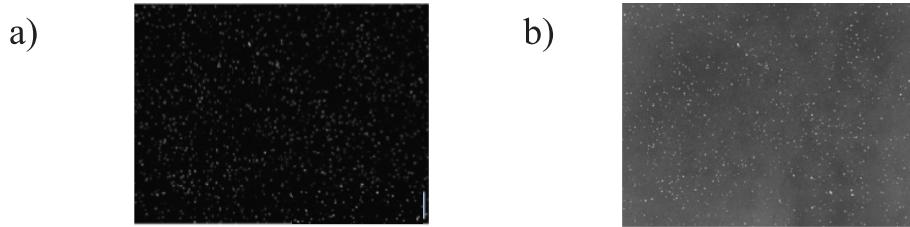
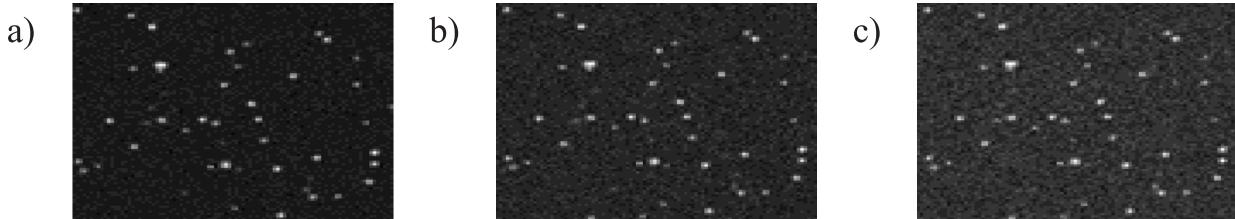
The framework and settings of the CNN were established using vortex images captured under uniform illumination conditions. We then compared the results obtained using the DCC method and various CNN architectures with the benchmark values. As shown in Table 3, Cases V1R1 and V1R2 with IAs of 64×64 presented no significant differences between the results obtained using the CNN and DCC methods and the VCC results; $\rho_v^2 = 2$ indicates values that are indistinguishable from the benchmark. Both CNN architectures presented promising results; however, Net 1 (with fewer convolution layers) extracted sufficient feature-related data for analysis. Note also that the benefits of adding convolutional layers diminished beyond a certain point, even as the computational overhead increased. Thus, we selected the Net-1 CNN architecture for subsequent analysis.

To assess the effects of activation functions, we compared Cases

Table 2

Imaging conditions in the simulations conducted.

Case No.	Flow fields	Seed	Seed diam.	Seed vel.	Illum.	GauNois	IA	Act. Func.	CNN frame
		den.							
V1R1	vortex	3 k	3	3	uniform	–	64	ReLU	Net 1
V1R2	vortex	3 k	3	3	uniform	–	64	ReLU	Net 2
V1LR1	vortex	3 k	3	3	uniform	–	64	LReLU	Net 1
V1PR1	vortex	3 k	3	3	uniform	–	64	PReLU	Net 1
V1EL1	vortex	3 k	3	3	uniform	–	64	ELU	Net 1
V1I32	vortex	3 k	3	3	uniform	–	32	ReLU	Net 1
V1I96	vortex	3 k	3	3	uniform	–	96	ReLU	Net 1
V1I128	vortex	3 k	3	3	uniform	–	128	ReLU	Net 1
V1R1N	vortex	3 k	3	3	non-uniform	–	64	ReLU	Net 1
V2R1	vortex	3 k	3	5	uniform	–	64	ReLU	Net 1
V2R1N	vortex	3 k	3	5	non-uniform	–	64	ReLU	Net 1
V3R1	vortex	10 k	3	3	uniform	–	64	ReLU	Net 1
V4R1	vortex	10 k	3	5	uniform	–	64	ReLU	Net 1
V1R1G1	vortex	3 k	3	3	uniform	15 %	64	ReLU	Net 1
V1R1G2	vortex	3 k	3	3	uniform	25 %	64	ReLU	Net 1
V1R1G3	vortex	3 k	3	3	uniform	35 %	64	ReLU	Net 1
J1R1	Jet	3 k	3	3	uniform	–	64	ReLU	Net 1
J1R1G1	Jet	3 k	3	3	uniform	15 %	64	ReLU	Net 1
J1R1G2	Jet	3 k	3	3	uniform	25 %	64	ReLU	Net 1
J1R1G3	Jet	3 k	3	3	uniform	35 %	64	ReLU	Net 1
U1R1	Uniform	3 k	3	3	uniform	–	64	ReLU	Net 1
U1R1G1	Uniform	3 k	3	3	uniform	15 %	64	ReLU	Net 1
U1R1G2	Uniform	3 k	3	3	uniform	25 %	64	ReLU	Net 1
U1R1G3	Uniform	3 k	3	3	uniform	35 %	64	ReLU	Net 1

**Fig. 7.** Image illumination conditions: a) uniform and b) non-uniform.**Fig. 8.** Signal interference in the form of Gaussian noise: a) 15%, b) 25%, and c) 35%.

V1R1, V1LR1, V1PR1, and V1EL1 (see [Table 2](#)) with various functions. Overall, the ReLU function outperformed the others by providing the lowest RMSE (compared to the benchmark) for the u or v velocity components (see [Fig. 10](#)). The ReLU function provides four advantages for deriving flow measurements. First, it is in line with the transmission of signals by biological neurons, which do not respond until a stimulus has reached a certain intensity and sends only weak signals when a given threshold is exceeded ([Glorot et al., 2011](#)). Second, in error transmission, the error gradient must be calculated in terms of the bias and transmitted back. Note that when the ReLU function is used for the output, the error gradient does not disappear. Third, the ReLU function causes the output of some neurons (those with input less than 0) to be zero, thereby increasing the sparseness of active neurons in the network, which is consistent with biological mechanisms ([Attwell and Laughlin, 2001; Glorot et al., 2011](#)). Finally, ReLU is simple to compute and imposes a reasonable computational overhead. Therefore, we used ReLU as

the activation function in the CNN architecture for all subsequent analyses.

The effect of IA was evaluated in Cases V1R1, V1I32, V1I96, and V1I128 (see [Table 2](#)). Our results revealed that the window size certainly affected the performance of the DCC and CNN (see [Fig. 11](#)). The appropriate size for the IA depends on the particle density, particle diameter, and flow pattern. The sensitivity of the DCC method to IA resulted in a wide range of measurement values. The CNN method was largely robust to variations in IA; however, the best results were obtained for a window size of 64×64 .

The proposed CNN was assessed using various architectures, activation functions, and window sizes. In the following sections, we consider a CNN with fewer convolution layers (Net-1), a ReLU active function, and a 64×64 IA.

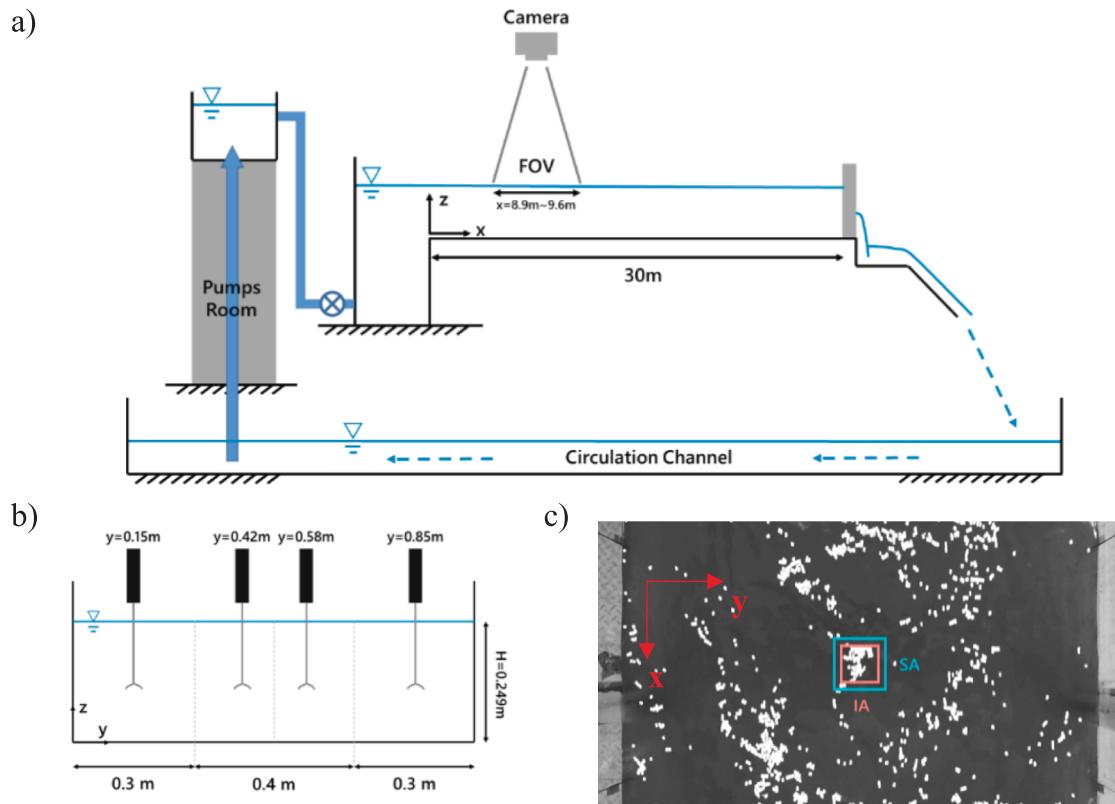


Fig. 9. Experimental setup for flume test: a) flume settings; b) ADV measurement positions; and c) IA and SA settings for imaging.

Table 3
VCC results for Net1 and Net2.

	Net1	Net2
CNN	1.9472	1.9463
DCC	1.9587	1.9587

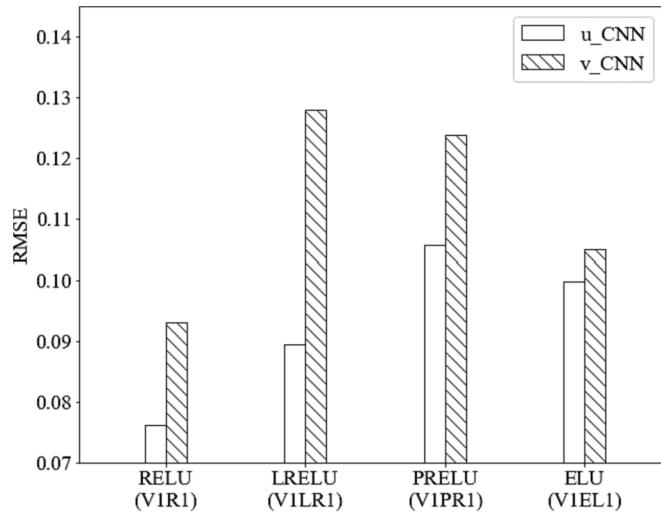


Fig. 10. Comparison of u and v velocity components as a function of four activation functions; u_CNN and v_CNN were used for CNN.

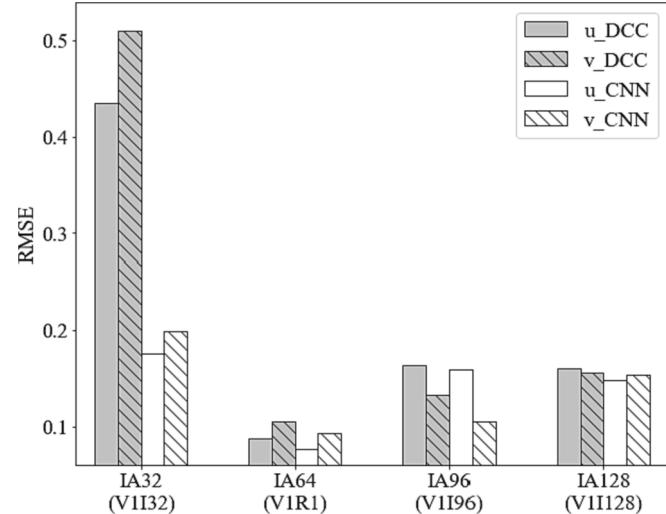


Fig. 11. Comparison of u and v velocity components using DCC and CNN with four IAs; u_DCC and v_DCC were used for DCC.

4.2. Impact of environmental noise on CNN and DCC

The illumination conditions and particle density are difficult to control in the field. Thus, we conducted experiments to evaluate the robustness of the CNN and DCC methods when using images captured under uniform and non-uniform illumination conditions. Case V1R1 was compared with Case V1R1N under a slow flow field, whereas case V2R1 was compared with Case V2R1N under a fast flow field; the results are shown in Fig. 12 and Table 4. Overall, the CNN generated similar results, regardless of illumination. Under uneven illumination, the DCC method

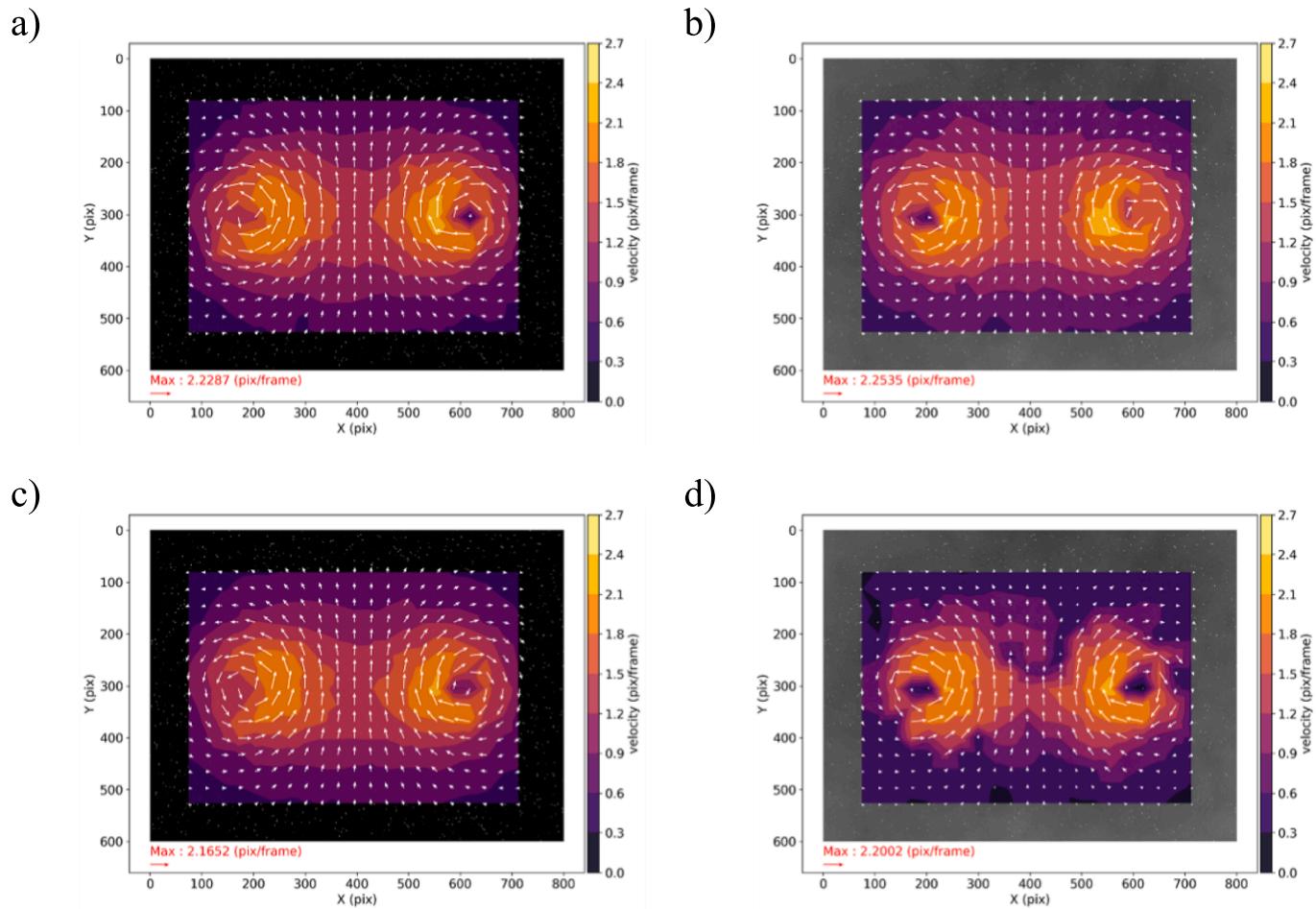


Fig. 12. Comparison of DCC and CNN results versus benchmark VCC values for a slow velocity field under illumination noise: a) V1R1; b) V1R1N; c) V1R1 for DCC; and d) V1R1N for DCC.

Table 4

Comparison of DCC and CNN results versus benchmark VCC values for a slow velocity field under uniform and non-uniform illumination conditions.

Uniform (V1R1)	Non-uniform (V1R1N)	Decrease
CNN	1.9472	1.9386
DCC	1.9587	1.8326

was able to capture the vortex but was unable to estimate the velocity magnitude with accuracy, particularly when the vectors were close to the center of the vortex. When the particle movement was increased (V2R1), the results obtained using the DCC method were profoundly affected by the illumination conditions (see Fig. 13). The DCC could barely capture the vortex, and the estimated velocities significantly differed from the benchmark values. When using the DCC method to analyze a slow flow field, the discrepancy between results obtained under uniform or non-uniform illumination was 6.43 %; however, when applied to a fast flow field, the discrepancy increased to 18.14 % (see Tables 4 and 5). These results indicate that the DCC method is strongly affected by image quality. The CNN method proved to be highly robust to image contamination. Our results indicate that the proposed CNN should be able to overcome the effects of ambient light noise, thereby capturing flow fields close to ground truth. However, under well-controlled lighting conditions, the CNN provided no advantage over the DCC method in deriving an accurate velocity field.

The effects of seeding density were assessed in Cases V1R1, V2R1, V3R1, and V4R1. Table 6 lists the measurements obtained using the

methods versus the benchmark values. A high seeding density increased the accuracy of the measurements, regardless of the computation method. Overall, DCC was more sensitive than CNN to seeding density; however, both methods provided reasonable measurements, regardless of seeding density.

4.3. Impact of Gaussian noise

Gaussian noise was used to simulate the noise in the images captured under low-light conditions. These results revealed a positive correlation between the amount of Gaussian noise and the discrepancy between the DCC velocity results and ground-truth values (see Table 7). The CNN method was significantly less sensitive to Gaussian noise. The addition of 35 % Gaussian noise increased the measurement error as follows: DCC, 2.77 to 31.13-fold increase; CNN, 1.25 to 1.68-fold increase.

Overall, the DCC method proved to be highly vulnerable to the imaging conditions, whereas the CNN method proved to be robust. Thus, it is reasonable to assume that the proposed CNN scheme outperforms the DCC method when it comes to overcoming environmental noise to obtain reliable flow measurements in practical situations.

4.4. Impact of flow field characteristics

The capability of the proposed CNN method was demonstrated in instances of uniform flow and jet flow, which are difficult to manage for the DCC method owing to problems associated with IA. As shown in Table 7, the CNN results were relatively stable despite the less

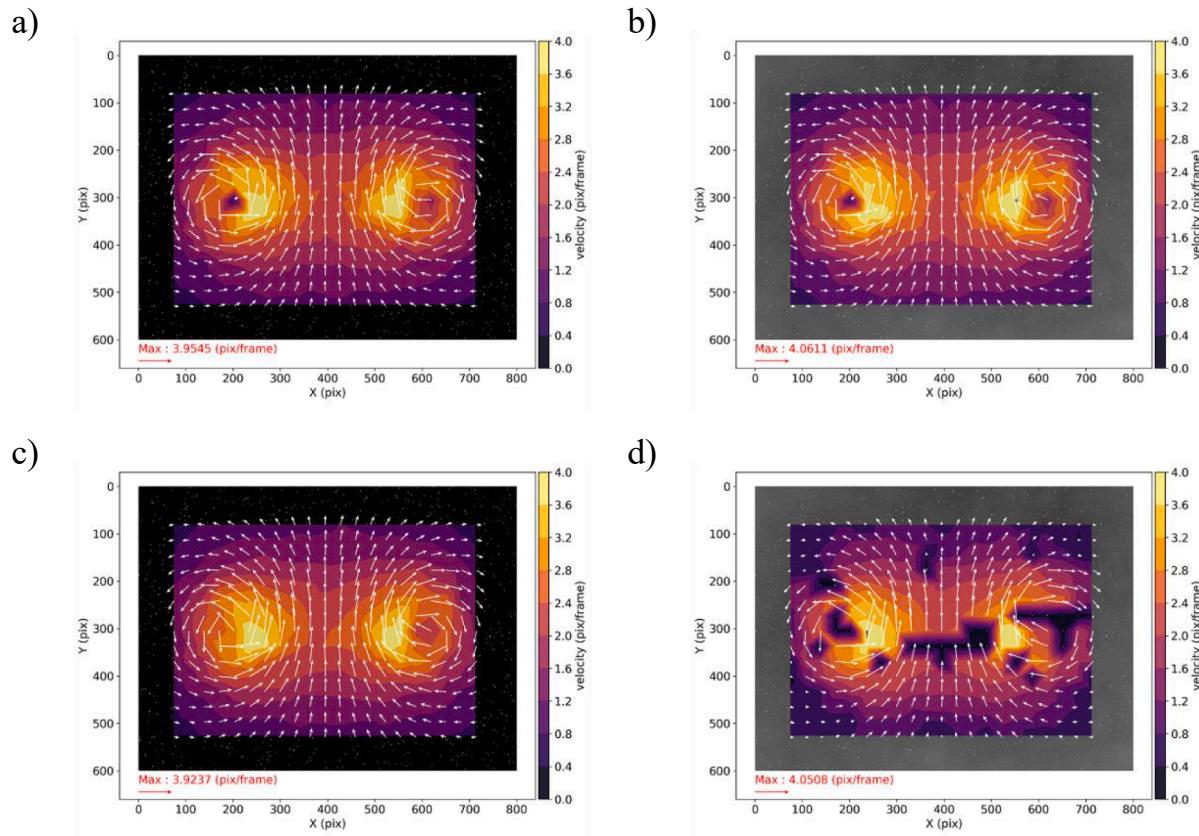


Fig. 13. Comparison of DCC and CNN results versus benchmark VCC values for a slow velocity field under illumination noise: a) V2R1; b) V2R1N; c) V2R1 for DCC; and d) V2R1N for DCC.

Table 5

Comparison of DCC and CNN results versus benchmark VCC values for a fast velocity field under uniform and non-uniform illumination conditions.

Uniform (V2R1)	Non-uniform (V2R1N)	Decrease
CNN	1.9283	1.9217
DCC	1.9148	1.5675

Table 6

Comparison of DCC and CNN results versus benchmark VCC values as a function of seeding density.

V1R1	V2R1	V3R1	V4R1
CNN	1.9472	1.9283	1.9667
DCC	1.9587	1.9148	1.9626

satisfactory performance in modeling jet flow. The tolerance for noise and IA makes the CNN method well suited for flow measurement under uncontrolled environmental conditions.

4.5. Computation time

Table 8 lists the computation times for all cases. Overall, the main factor affecting the computation time was the size of the IA. The computational overhead for the proposed CNN scheme was insufficient to counterbalance the advantages. In fact, the proposed CNN should be able to provide results in real time.

4.6. Flume experiments

Flume experiments were performed to assess the efficacy of the CNN

Table 7

RMSE for vortex, jet flow, and uniform flow obtained using DCC and CNN methods under Gaussian noise.

Vortex flow				
	V1R1	V1R1G1	V1R1G2	V1R1G3
u_DCC	0.0874	0.1188	0.2279	0.4478
u_CNN	0.0918	0.0961	0.1059	0.1331
v_DCC	0.1049	0.1392	0.2900	0.5959
v_CNN	0.1141	0.1208	0.1317	0.1466
Jet flow				
	J1R1	J1R1G1	J1R1G2	J1R1G3
u_DCC	0.3496	0.3797	0.5314	0.9701
u_CNN	0.2576	0.2632	0.2882	0.3214
v_DCC	0.0421	0.0635	0.1093	0.1374
v_CNN	0.0467	0.0527	0.0678	0.0720
Uniform flow				
	U1R1	U1R1G1	U1R1G2	U1R1G3
u_DCC	0.0208	0.0219	0.0266	0.4628
u_CNN	0.0334	0.0400	0.0493	0.0560
v_DCC	0.0165	0.0193	0.0228	0.5136
v_CNN	0.0362	0.0420	0.0455	0.0475

and DCC methods, and the results were compared with measurements obtained using ADV. Instantaneous velocities were derived using 200 successive images. The statistical dispersion of these velocities is illustrated in **Fig. 14**, which shows the velocity components in the x and y directions. The instantaneous velocities obtained using the DCC method included a large number of outliers, indicating evident errors (in the

Table 8
Running time for all cases.

Case No.	Runtime (sec)	Case No.	Runtime (sec)
V1R1	103.86	V4R1	102.64
V1R2	102.83	V1R1G1	102.82
V1LR1	104.47	V1R1G2	102.81
V1PR1	103.70	V1R1G3	101.84
V1EL1	102.85	J1R1	106.53
V1I32	368.62	J1R1G1	103.41
V1I96	71.34	J1R1G2	101.82
V1I128	24.36	J1R1G3	101.50
V1R1N	106.35	U1R1	104.16
V2R1	101.88	U1R1G1	103.42
V2R1N	104.38	U1R1G2	102.34
V3R1	100.86	U1R1G3	104.87

opposite direction). The CNN method exhibited stable performance, presenting only one outlier. Estimates of the discharge from the surface flow measurements were derived from the mean velocities. In the flume experiment, the main flow ran in the x direction, such that the average velocity of only the u component was used to compare both methods. To improve the DCC results, we eliminated outlier measurements larger than their two corresponding standard deviations, and manually deleted the opposite velocity values before deriving average surface velocities. Once these outliers were removed, the CNN and DCC methods presented similar results in terms of vector fields and velocity magnitude contours

(see Fig. 15). Overall, the flume experiment corroborated the capability of the CNN method to overcome low seeding density and surface reflection induced by standing waves in the flume.

The surface velocity was estimated from ADV measurements by fitting vertical ADV data to log-law profiles at positions $x = 9.27$ m and $y = 0.1, 0.42, 0.58$, and 0.85 m. The R^2 value for all positions was 0.94. At least 5 min of data measurements were collected for each point under a signal-to-noise ratio larger than 80 %. For each position, the surface and depth-averaged velocities were calculated from the fitted vertical velocity profile function; the velocity index was estimated using the depth-averaged velocity divided by the surface velocity (see Table 9). Using the average velocity, it was possible to estimate the discharge in each zone to obtain a total discharge of $0.0752 \text{ m}^3/\text{s}$ for the cross-section at $x = 9.27$ m.

CNN and DCC estimates of discharge were obtained by multiplying surface flow velocities at positions $x = 9.27$ m and $y = 0.15, 0.42, 0.58$, and 0.85 m by the corresponding velocity index derived from ADV measurements in order to obtain the depth-averaged unit width flow-rate, which was then multiplied by the respective cross-section areas to obtain the total flow rate. CNN and DCC provided similar estimates with an error ranging from 7.87 to 8.09 % with respect to ADV measurements (see Table 10).

In this experiment, the illumination was not controlled using an additional light source. However, the brightness distribution on the water surface was fairly uniform, and the color of the image background

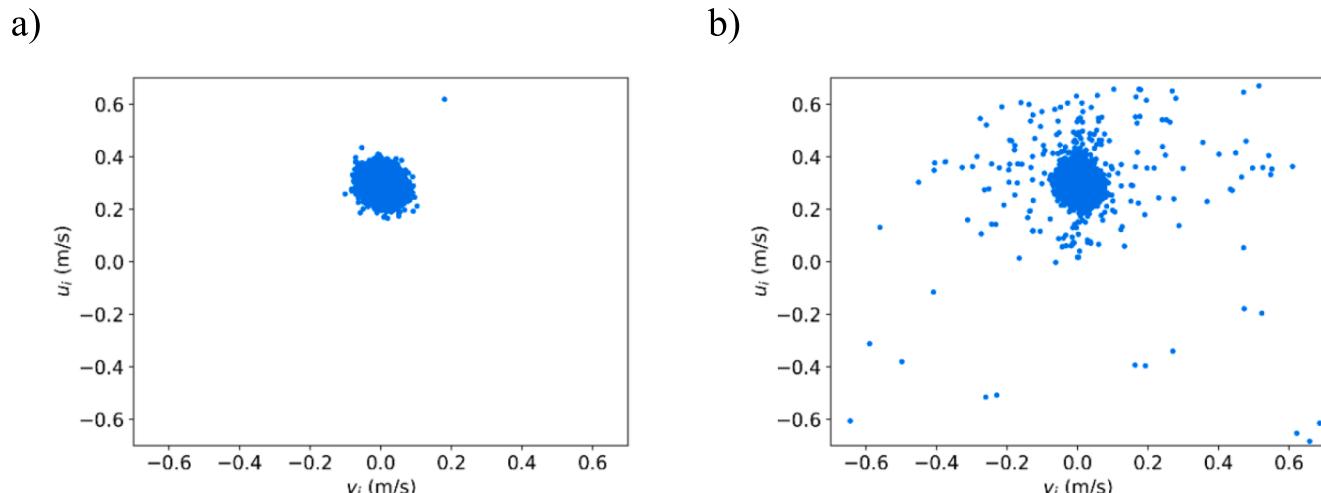


Fig. 14. Distribution of instantaneous velocities (u and v) measured using a) CNN and b) DCC.

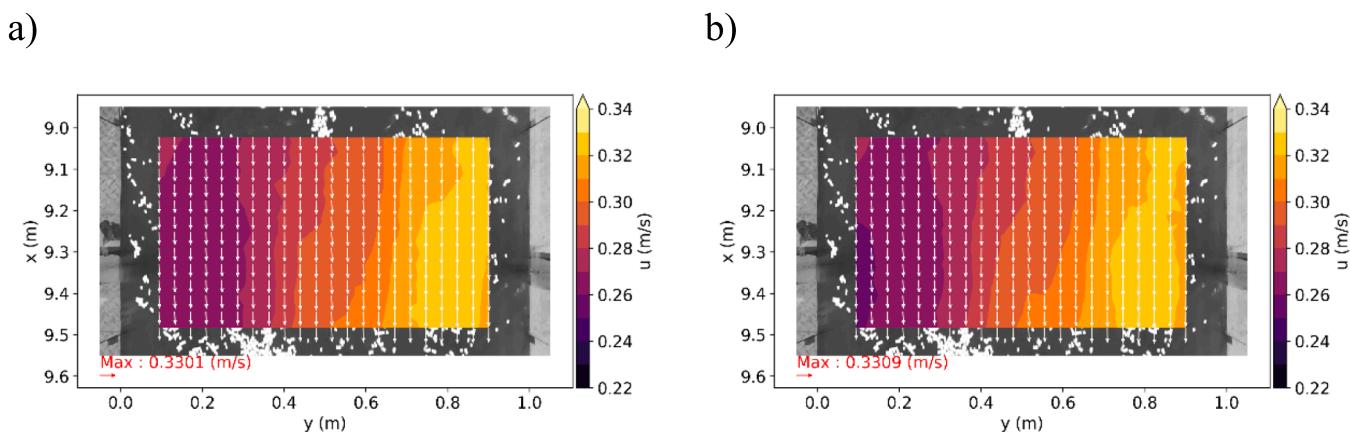


Fig. 15. Average surface velocity contours of us (x, y) obtained using a) CNN and b) DCC.

Table 9

Average surface velocity u_s , depth-averaged velocity \bar{u} , and velocity index α calculated from ADV measurements.

y (m)	0.15	0.42	0.58	0.85
\bar{u} (m/s)	0.3023	0.2963	0.3013	0.3062
u_s (m/s)	0.3145	0.3128	0.3199	0.3214
α	0.9611	0.9473	0.9420	0.9525
Q_y (cms)	0.0226	0.0148	0.015	0.0229
Q_{total} (cms)	0.0752			

Table 10

Total discharge estimated using CNN, DCC, and ADV.

	Q_{total} (cms)	Error with ADV
ADV	0.0752	
DCC	0.0691	8.09 %
CNN	0.0693	7.87 %

was distinctly different from that of the particles. The diameter of the particles in the analyzed images was approximately 10 px, and the particle density was low, similar to the artificial images previously discussed. Despite the fact that the imaging conditions were not precisely the same as those of the simulations, the same conclusions (in terms of velocity measurement) can be drawn. The robustness of the proposed CNN method made it possible to obtain reliable measurements of velocity, which makes it a viable alternative to current techniques used in the field.

5. Conclusions

Imaging technology is a highly useful, non-intrusive approach for flow measurement based on surface velocity fields. In other words, imaging technology makes it easier to achieve large-scale velocity measurements at high sampling rates. Nonetheless, image noise introduced by environmental conditions (e.g., variable illumination and low-light artifacts), seeding distribution, and the effects of image-matching algorithms can have a profound impact on flow measurements. The CNN method proposed in this study is a feasible approach for obtaining flow measurements in the field using conventional imaging techniques, such as LSPIV.

Velocity measurements obtained using successive artificial images revealed that non-uniform illumination can significantly degrade image

matching; however, the CNN method proved to be robust to this type of environmental noise. Non-uniform illumination was shown to reduce the accuracy of DCC measurements by 18.14 % while reducing the accuracy of CNN measurements by only 0.34 %. The DCC method could not delineate the flow features of the vortex. We also examined the influence of Gaussian noise on various flow fields (vortices, jets, and uniform flows). The addition of 35 % Gaussian noise increased the measurement error (in terms of RMSE) of the DCC by 2.77 to 31.13-fold, whereas it increased the error of CNN by only 1.25 to 1.68-fold.

Flume experiments were also conducted to assess the capability of the CNN and DCC in estimating discharge. When imaging parameters were well controlled and outliers were filtered out, both methods performed similarly, diverging from ground-truth values (ADV) in the range of 7.87 to 8.09 % for discharge. When measuring instantaneous velocity, the proposed CNN scheme generated far fewer outliers. The network architecture, activation functions, and depth of the convolution layer can significantly affect the estimation results. Here, we opted for a two-layer framework in conjunction with a convolutional layer [16,32], fully-connected layer [4096], and ReLU for flow measurement.

The robustness of the proposed CNN method makes it possible to obtain reliable measurements of velocity under conditions of high environmental noise, thereby demonstrating its viability as an alternative to current field techniques.

CRediT authorship contribution statement

Hao-Che Ho: Conceptualization, Methodology, Software, Formal analysis, Investigation, Resources, Writing – original draft, Writing – review & editing. **Yu-Wei Chiu:** Software, Validation, Investigation, Visualization. **Ting-Yu Chen:** Software, Investigation, Data curation. **Yen-Cheng Lin:** Data curation, Writing – review & editing, Visualization.

Declaration of Competing Interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Hao-Che Ho reports financial support was provided by National Science and Technology Council.

Data availability

Data will be made available on request.

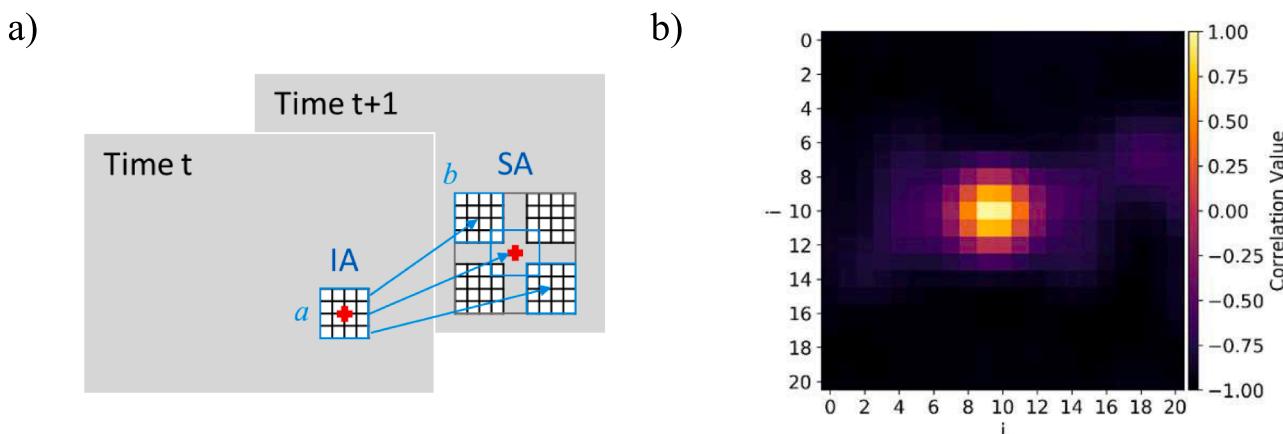


Fig. A.1. (a) Definition of IA and SA, and (b) contour of cross-correlation matrix R_g .

Appendix A

The DCC method calculates the correlation coefficient (r_{ab}) between the IA at time t and the intensity of all possible IAs in the SA at time $t + 1$ (see Fig. A.1a) using the following equation (Fujita et al., 1998):

$$r_{ab} = \frac{\sum_{i=1}^M \sum_{j=1}^N (a_{ij} - \bar{a}_{ij})(b_{ij} - \bar{b}_{ij})}{\sqrt{\sum_{i=1}^M \sum_{j=1}^N (a_{ij} - \bar{a}_{ij})^2 \sum_{i=1}^M \sum_{j=1}^N (b_{ij} - \bar{b}_{ij})^2}} \quad (\text{A.1})$$

where i and j represent the coordinates of the elements in image matrices a and b , respectively.

Once the correlation coefficients of sub-image a and all candidate sub-images b are calculated, the correlation coefficient matrix (R_g) is obtained using the above equation for sub-image a at a fixed grid point g between the images at times t and $t + 1$. Generally, if the image quality is suitably controlled, then there should be only one significant peak (z_{\max}) (see Fig. A.1b). Thus, z_{\max} corresponding to sub-image b is defined as the one that matches sub-image a at $t + 1$, and the difference between the positions of both sub-images (a and b) is obtained as a movement vector $[\delta x_g, \delta y_g]$ with pixel-level accuracy at grid point g . The velocity is then calculated in accordance with this movement vector.

Appendix B

We assessed the effects of four activation functions: the rectified linear unit (ReLU), leaky ReLU (LReLU), parametric rectified linear unit (PReLU), and exponential linear unit (ELU), as shown in Fig. B.1. ReLU presents no saturation region or gradient dissipation, and its computational overhead is low (Nair and Hinton, 2010). Furthermore, feedback is provided only when $x > 0$, which is in line with the activation mechanism of biological nerves, wherein a neuron is deactivated when $x < 0$ to avoid updating parameters during training. ReLU requires a small gradient assigned by a fixed value that is normally smaller than zero. PReLU requires a variable (α), which is learned from the slope of the data when $x < 0$ (He et al., 2015). If α is small, then the formula is the same as that of LReLU, whereas if $\alpha = 0$, then the formula will be the same as that of ReLU. PReLU has been shown to approach human classification ability in the ImageNet 2012 classification dataset (He et al., 2015; Krizhevsky et al., 2017). Compared to ReLU, ELU requires an extra constant α to define the smoothness function when the inputs are negative. By reducing the bias shift effect, ELU makes it possible to reduce the training time and enhance accuracy (Clevert et al., 2015).

Appendix C

The covariance matrix between two vector fields is given by the following equation (Crosby et al., 1993):

$$\Sigma_W = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}, \quad (\text{C.1})$$

where

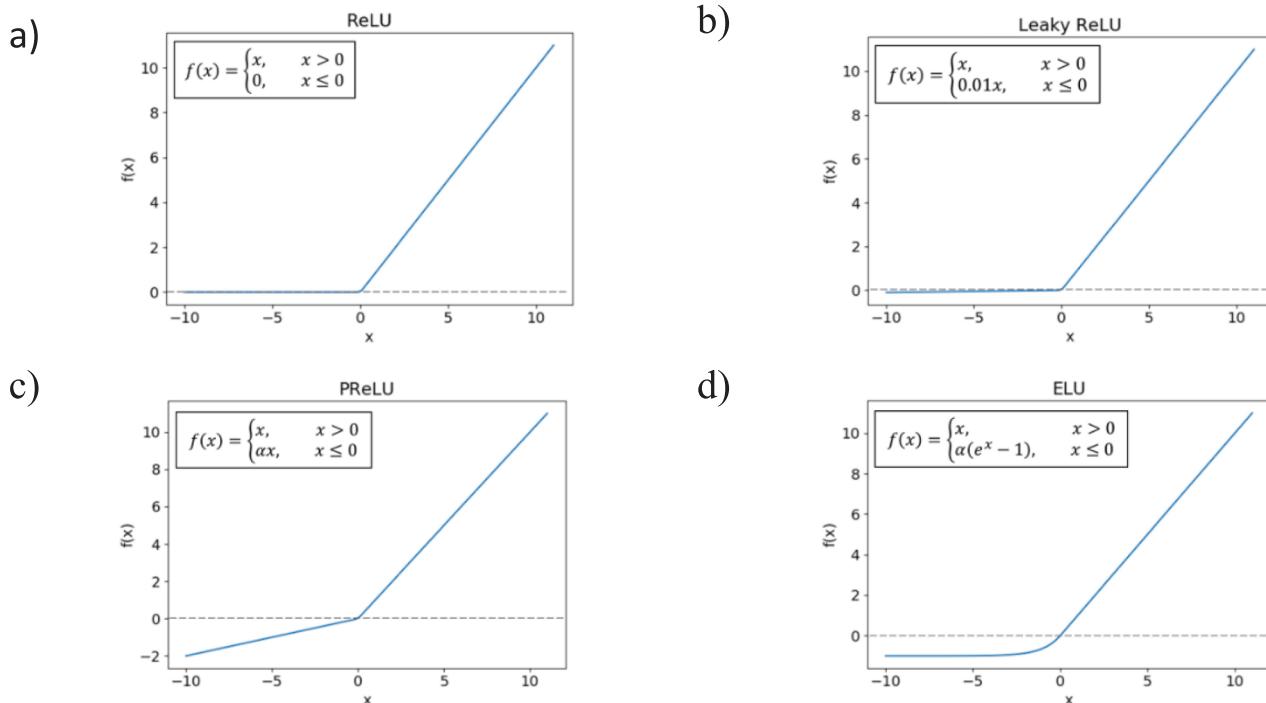


Fig. B.2. Activation functions in CNN networks: a) ReLU; b) LReLU; c) PReLU; and d) ELU.

$$\Sigma_{11} = \begin{bmatrix} \sigma(u_1, u_1) & \sigma(u_1, v_1) \\ \sigma(v_1, u_1) & \sigma(v_1, v_1) \end{bmatrix}, \quad (C.2)$$

$$\Sigma_{12} = \begin{bmatrix} \sigma(u_1, u_2) & \sigma(u_1, v_2) \\ \sigma(v_1, u_2) & \sigma(v_1, v_2) \end{bmatrix}, \quad (C.3)$$

$$\Sigma_{21} = \begin{bmatrix} \sigma(u_2, u_1) & \sigma(u_2, v_1) \\ \sigma(v_2, u_1) & \sigma(v_2, v_1) \end{bmatrix}, \quad (C.4)$$

$$\Sigma_{22} = \begin{bmatrix} \sigma(u_2, u_2) & \sigma(u_2, v_2) \\ \sigma(v_2, u_2) & \sigma(v_2, v_2) \end{bmatrix}. \quad (C.5)$$

The VCC (ρ_v^2) is calculated as follows:

$$\rho_v^2 = Tr[(\Sigma_{11})^{-1}\Sigma_{12}(\Sigma_{22})^{-1}\Sigma_{21}], \quad (C.6)$$

where Tr is the sum of matrix diagonal elements.

The calculation of VCC can be facilitated by expanding Eq. (C.6) as follows:

$$\rho_v^2 = \frac{f}{g}, \quad (C.7)$$

where

$$\begin{aligned} f = & \sigma(u_1, v_1)\{\sigma(u_2, v_2)[\sigma(v_1, v_2)]^2 + \sigma(v_2, v_2)[\sigma(v_1, u_2)]^2\} \\ & + \sigma(v_1, v_1)\{\sigma(u_2, v_2)[\sigma(u_1, v_2)]^2 + \sigma(v_2, v_2)[\sigma(u_1, u_2)]^2\} \\ + & 2[\sigma(u_1, v_1)\sigma(u_1, v_2)\sigma(v_1, u_2)\sigma(u_2, v_2)] + 2[\sigma(u_1, v_1)\sigma(u_1, u_2)\sigma(v_1, v_2)\sigma(u_2, v_2)], \\ - & 2[\sigma(u_1, u_1)\sigma(v_1, u_2)\sigma(v_1, v_2)\sigma(u_2, v_2)] - 2[\sigma(v_1, v_1)\sigma(u_1, u_2)\sigma(u_1, v_2)\sigma(u_2, v_2)] \\ - & 2[\sigma(u_2, u_2)\sigma(u_1, v_1)\sigma(v_1, v_2)] - 2[\sigma(v_2, v_2)\sigma(u_1, v_1)\sigma(u_1, u_2)\sigma(v_2, v_2)] \end{aligned}$$

$$g = \{\sigma(u_1, u_1)\sigma(v_1, v_1) - [\sigma(u_1, v_1)]^2\}\{\sigma(u_2, u_2)\sigma(v_2, v_2) - [\sigma(u_2, v_2)]^2\}.$$

References

- Adrian, R.J., 1991. Particle-imaging techniques for experimental fluid mechanics. *Annu. Rev. Fluid Mech.* 23 (1), 261–304.
- Attwell, D., Laughlin, S.B., 2001. An energy budget for signaling in the grey matter of the brain. *J. Cereb. Blood Flow Metab.* 21 (10), 1133–1145.
- Bradley, A.A., Kruger, A., Meselte, E.A., Muste, M.V., 2002. Flow measurement in streams using video imagery. *Water Resour. Res.* 38 (12), 51–61.
- Capart, H., Young, D.-L., Zech, Y., 2002. Voronoi imaging methods for the measurement of granular flows. *Exp. Fluids* 32 (1), 121–135.
- Chaves, H., 2012. *A weather independent illumination for field LSPIV*. In: 16th Int Symp on Applications of Laser Techniques to Fluid Mechanics, Lisbon, Portugal.
- Chiang, Y.-M., Hao, R.N., Ho, H.-C., Chang, T.-J., Xu, Y.P., 2017. Evaluating the contribution of multi-model combination to streamflow hindcasting by empirical and conceptual models. *Hydrol. Sci. J.* 62 (9), 1456–1468.
- Clevert, D. A., Unterthiner, T., Hochreiter, S., 2015. *Fast and accurate deep network learning by exponential linear units (elus)*. arXiv preprint arXiv:1511.07289.
- Creutin, J.D., Muste, M., Li, Z., 2002. Traceless quantitative imaging alternatives for free-surface measurements in natural stream. *Hydraulic Measurements and Experimental Methods Specialty Conference (HMEM)*, Estes Park, Colorado, United States. Estes Park, Colorado, United States.
- Crosby, D.S., Brecker, L.C., Gemmill, W.H., 1993. A proposed definition for vector correlation in geophysics: Theory and application. *J. Atmos. Ocean. Technol.* 10 (3), 355–367.
- De Boer, P.T., Kroese, D.P., Mannor, S., Rubinstein, R.Y., 2005. A tutorial on the cross-entropy method. *Ann. Oper. Res.* 134 (1), 19–67.
- Deng, H., Chen, W., Huang, G., 2022. Deep insight into daily runoff forecasting based on a CNN-LSTM model. *Nat. Hazards* 113 (3), 1675–1696.
- Duchi, J., Hazan, E., Singer, Y., 2011. Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn. Res.* 12 (7), 2121–2159.
- Forghani, M., Qian, Y., Lee, J., Farthing, M.W., Hesser, T., Kitaniidis, P.K., Darve, E.F., 2021. Application of deep learning to large scale riverine flow velocity estimation. *Stoch. Env. Res. Risk A* 35 (5), 1069–1088.
- Fujita, I., Komura, S., 1994. Application of video image analysis for measurements of river-surface flows. *Proc. Hydraul. Eng.* 38, 733–738.
- Fujita, I., Muste, M., Kruger, A., 1998. Large-scale particle image velocimetry for flow analysis in hydraulic engineering applications. *J. Hydraul. Res.* 36 (3), 397–414.
- Ghashghaei, M., Eslami, H., Ostad-Ali-Askari, K., 2022. Applications of time series analysis to investigate components of Madiyan-rood river water quality. *Appl. Water Sci.* 12 (8), 202.
- Glorot, X., Bengio, Y., 2010. *Understanding the difficulty of training deep feedforward neural networks*. In: Proceeding of the thirteenth international conference on artificial intelligence and statistics, Chia Laguna Resort, Sardinia, Italy.
- Glorot, X., Bordes, A., Bengio, Y., 2011. Deep sparse rectifier neural network. In: Lauderdale, F. (Ed.), *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, Fort Lauderdale, Florida, United States.
- Hann, D.B., Greated, C.A., 1997. The measurement of flow velocity and acoustic particle velocity using particle-image velocimetry. *Meas. Sci. Technol.* 8 (12), 1517.
- Hao, W., Bie, R., Guo, J., Meng, X., Wang, S., 2018. Optimized CNN based image recognition through target region selection. *Optik* 156, 772–777.
- Hauet, A., Creutin, J.D., Belleudy, P., 2008. Sensitivity study of large-scale particle image velocimetry measurement of river discharge using numerical simulation. *J. Hydrol.* 349 (1–2), 178–190.
- He, K., Zhang, X., Ren, S., Sun, J., 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *IEEE International Conference on Computer Vision (ICCV)*, Santiago Chile.
- Hinton, G. E. 1986. *Learning distributed representations of concepts*. In: *Proceedings of the Eighth Annual Conference of the Cognitive Science Society*. Amherst, Massachusetts, United States.
- Ioffe, S., Szegedy, C., 2015. *Batch normalization: Accelerating deep network training by reducing internal covariate shift*. In: *Proceedings of the 32nd International Conference on Machine Learning*, Lille, France.
- Jähne, B., 1995. Concepts, algorithms, and scientific applications. In: *Digital Image Processing*. Springer-Verlag, pp. 321–323.
- Kagaya, H., Aizawa, K., Ogawa, M., 2014. *Food detection and recognition using convolutional neural network*. In: *Proceedings of the 22nd ACM International Conference on Multimedia*, Orlando, Florida, United States.
- Karpatne, A., Kannan, R., Kumar, V. (Eds.), 2022. *Knowledge Guided Machine Learning: Accelerating Discovery using Scientific Knowledge and Data*. CRC Press.
- Kim, D., Ho, H.-C., Baranya, S., Muste, M., 2016. Qualitative and quantitative acoustic mapping of bedform dynamics. *Flow Meas. Instrum.* 50, 80–89.
- Kingma, D. P., Ba, J. 2014. *Adam: A method for stochastic optimization*. arXiv preprint arXiv:1412.6980.
- Kouadri, S., Elbeltagi, A., Islam, A.R.M.T., Kateb, S., 2021. Performance of machine learning methods in predicting water quality index based on irregular data set: application on Illizi region (Algerian southeast). *Appl. Water Sci.* 11 (12), 190.
- LeCun, Y., Bottou, L., Bengio, Y., Haffner, P. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86(11), 2278–2324.
- Krizhevsky, A., Sutskever, I., Hinton, G.E., 2017. *Imagenet classification with deep convolutional neural networks*. *Communications of the ACM* 60 (6), 84–90.

- Lipper, L., McCarthy, N., Zilberman, D., Asfaw, S., Branca, G., 2017. Climate Smart Agriculture: Building Resilience to Climate Change. Springer Nature, p. 630.
- Ma, J., Jiang, X., Fan, A., Jiang, J., Yan, J., 2021. Image matching from handcrafted to deep features: A survey. *Int. J. Comput. Vis.* 129 (1), 23–79.
- Makino, S., Kawabata, T., Kido, K. i. 1983. *Recognition of consonant based on the perceptron model*. In: ICASSP'83. IEEE International Conference on Acoustics, Speech, and Signal Processing, Boston, Massachusetts, United States.
- Morgan, N., Bourlard, H. 1990. *Continuous speech recognition using multilayer perceptrons with hidden Markov models*. In: International conference on acoustics, speech, and signal processing, Albuquerque, New Mexico, United States.
- Muste, M., Hauet, A., Fujita, I., Legout, C., Ho, H.-C., 2014. Capabilities of large-scale particle image velocimetry to characterize shallow free-surface flows. *Adv. Water Resour.* 70, 160–171.
- Nair, V., Hinton, G.E., 2010. Rectified linear units improve restricted Boltzmann machines. International Conference on Machine Learning, Haifa, Israel.
- Nobach, H., Honkanen, M., 2005. Two-dimensional Gaussian regression for sub-pixel displacement estimation in particle image velocimetry or particle position estimation in particle tracking velocimetry. *Exp. Fluids* 38 (4), 511–515.
- Ostad-Ali-Askari, K., Shayan, M., 2021. Subsurface drain spacing in the unsteady conditions by HYDRUS-3D and artificial neural networks. *Arab. J. Geosci.* 14 (18), 1–14.
- Ostad-Ali-Askari, K., Shayannejad, M., 2021. Computation of subsurface drain spacing in the unsteady conditions using Artificial Neural Networks (ANN). *Appl. Water Sci.* 11 (2), 21.
- Ostad-Ali-Askari, K., Shayannejad, M., Ghorbanizadeh-Kharazi, H., 2017. Artificial neural network for modeling nitrate pollution of groundwater in marginal area of Zayandeh-rood River, Isfahan, Iran. *KSCE J. Civil Eng.* 21 (1), 134–140.
- Özyurt, F., 2020. Efficient deep feature selection for remote sensing image recognition with fused deep learning architectures. *J. Supercomput.* 76 (11), 8413–8431.
- Qian, N., 1999. On the momentum term in gradient descent learning algorithms. *Neural Netw.* 12 (1), 145–151.
- Rosenblatt, F., 1958. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychol. Rev.* 65 (6), 386.
- Saffman, P.G., 1992. Vortex Dynamics. Cambridge University Press.
- Schlichting, V.H., 1933. Laminare strahlausbreitung. *ZAMM-J. Appl. Math. Mech./Zeitschrift für Angewandte Mathematik und Mechanik* 13 (4), 260–263.
- Shang, L., Yang, Q., Wang, J., Li, S., Lei, W., 2018. *Detection of rail surface defects based on CNN image recognition and classification*. 20th International Conference on Advanced Communication Technology (ICACT), Chuncheon-si, Gangwon-do, South Korea.
- Stamhuis, E.J., 2006. *Basics and principles of particle image velocimetry (PIV) for mapping biogenic and biologically relevant flows*. *Aquat. Ecol.* 40 (4), 463–479.
- Thielicke, W., Stamhuis, E.J., 2014. PIVlab – towards user-friendly, affordable and accurate digital particle image velocimetry in MATLAB. *J. Open Res. Softw.* 2 (1), 30.
- Traore, B.B., Kamsu-Foguem, B., Tangara, F., 2018. Deep convolution neural network for image recognition. *Eco. Inform.* 48, 257–268.
- Watrous, R.L., Shastri, L., 1987. Learning phonetic features using connectionist networks. *J. Acoust. Soc. Am.* 81 (1), S93–S94.
- Willert, C.E., Gharib, M., 1991. Digital particle image velocimetry. *Exp. Fluids* 10 (4), 181–193.
- Xiong, J., Yu, D., Liu, S., Shu, L., Wang, X., Liu, Z., 2021. A review of plant phenotypic image recognition technology based on deep learning. *Electronics* 10 (1), 81.
- Xu, M., Sun, Q., Huang, C., Shi, J., 2017. Object motion detection and data processing in large-scale particle image velocimetry. *Intell. Autom. Soft Comput.* 23 (4), 653–660.
- Young, D.S., Hart, J.K., Martinez, K., 2015. Image analysis techniques to estimate river discharge using time-lapse cameras in remote locations. *Comput. Geosci.* 76, 1–10.
- Zhang, Z., Wang, X., Fan, T., Xu, L., 2013. River surface target enhancement and background suppression for unseeded LSPIV. *Flow Meas. Instrum.* 30, 99–111.
- Zhen, Z., Yang, Z., Yuchou, L., Youjie, Y., Xurui, L. 2017. *IP camera-based LSPIV system for on-line monitoring of river flow*. In: 13th IEEE International Conference on Electronic Measurement & Instruments (ICEMI), Yangzhou, China.