

Water Resources Research



RESEARCH ARTICLE

10.1029/2019WR024833

Special Section:

Big Data & Machine Learning in Water Sciences: Recent Progress and Their Use in Advancing Science

Key Points:

- A convolutional neural network is trained on density-driven solute transport simulations to incorporate the physical process representation
- Missing velocity fields on experimental data are estimated utilizing the physical process representation learned on purely synthetic data
- Light transmission measurements using a Hele-Shaw cell are conducted to infer spatially resolved solute concentration distributions

Correspondence to:

P. J. Kreyenberg,
philipp.kreyenberg@iup.uni-heidelberg.de

Citation:

Kreyenberg, P. J., Bauser, H. H., & Roth, K. (2019). Velocity field estimation on density-driven solute transport with a convolutional neural network. *Water Resources Research*, 55, 7275–7293. <https://doi.org/10.1029/2019WR024833>

Received 23 JAN 2019

Accepted 23 JUL 2019

Accepted article online 1 AUG 2019

Published online 29 AUG 2019

©2019. The Authors.

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

Velocity Field Estimation on Density-Driven Solute Transport With a Convolutional Neural Network

Philipp J. Kreyenberg¹, Hannes H. Bauser^{1,2}, and Kurt Roth^{1,3}

¹Institute of Environmental Physics (IUP), Heidelberg University, Heidelberg, Germany, ²Biosphere 2, University of Arizona, Tucson, AZ, USA, ³Interdisciplinary Center for Scientific Computing (IWR), Heidelberg University, Heidelberg, Germany

Abstract Recent advances in machine learning open new opportunities to gain deeper insight into hydrological systems, where some relevant system quantities remain difficult to measure. We use deep learning methods trained on numerical simulations of the physical processes to explore the possibilities of closing the information gap of missing system quantities. As an illustrative example we study the estimation of velocity fields in numerical and laboratory experiments of density-driven solute transport. Using high-resolution observations of the solute concentration distribution, we demonstrate the capability of the method to structurally incorporate the representation of the physical processes. Velocity field estimation for synthetic data for both variable and uniform concentration boundary conditions showed equal results. This capability is remarkable because only the latter was employed for training the network. Applying the method to measured concentration distributions of density-driven solute transport in a Hele-Shaw cell makes the velocity field assessable in the experiment. This assessability of the velocity field even holds for regions with negligible solute concentration between the density fingers, where the velocity field is otherwise inaccessible.

1. Introduction

Gaining a quantitative understanding of hydrological systems is difficult and relies on the availability of accurate measurements that are dense in space and time. More and more of such data become available with increasing deployment of sensors, for example, satellite-based observations or embedded sensor networks. However, some relevant system quantities, for instance, local flow velocities, remain difficult to measure. On the other hand, simulations offer the advantage of detailed information, also of the quantities that are difficult to measure. They are often based on a good physical understanding, but the presence of nonlinear processes and multiscale heterogeneities typically impedes accurate prediction (e.g., Clark et al., 2011; Nowak & Cirpka, 2006; Vogel et al., 2018).

Advances can be made by closing the information gap of missing system quantities with consistent information transfer from simulation of relevant physical processes to the real world. Associated with this is the evaluation of the representation of relevant physical processes in the simulation. As elucidated in Marçais and de Druzy (2017), Shen (2018), and Shen et al. (2018), the progress and increasing availability of modern deep learning algorithms combined with the increasing availability of measured data open new possibilities to address these challenges.

To explore these possibilities, we focus on a comparably simple problem as an example: velocity field estimation on density-driven active solute transport observed in a small-scale laboratory experiment within a Hele-Shaw cell, where high-resolution measurements of the solute concentration distribution are available.

Density-driven active solute transport is a relevant process for geological storage of anthropogenic CO₂ (Ennis-King & Paterson, 2003, 2005; Lindeberg & Wessel-Berg, 1997; Weir et al., 1995). Capturing atmospheric CO₂ and storing it in 1- to 3-km-deep geological brine formations is, among others, one proposed technique to mitigate climate change (IPCC, 2005). At the prevalent conditions in these depths the supercritical CO₂, being trapped underneath impermeable cap rock, overlies the resident brine. The CO₂ dissolves into the brine leading to a local density increase at the interface. Eventually, this gives rise to density-driven instabilities drastically shortening the time scale of the mixing process in contrast to pure diffusion (Ennis-King & Paterson, 2003; Farajzadeh et al., 2007; Hassanzadeh et al., 2005; Kneafsey & Pruess, 2010;

Pau et al., 2010; Pruess & Zhang, 2008; Yang & Gu, 2006). Density-driven flow is a key process in several other settings beyond CO₂ sequestration. Examples include the description of water dynamics beneath saline lake formations (Wooding, Tyler, & White, 1997; Wooding, Tyler, White, & Anderson, 1997), toxic and radioactive waste disposal (Kolditz et al., 1998), and saltwater intrusion into exploited coastal aquifers (Diersch & Kolditz, 2002).

To investigate the dynamics of density-driven flow, several experimental studies using optical observation of CO₂ and brine analogous solutions in Hele-Shaw cells have been conducted at the laboratory scale (Backhaus et al., 2011; Ecke & Backhaus, 2016; Faisal et al., 2013, 2015; Fernandez et al., 2002; Kneafsey & Pruess, 2010, 2011; Oltean et al., 2004; Rasmusson et al., 2017; Slim et al., 2013; Thomas et al., 2018). As shown by Thomas et al. (2015), the use of color indicators often fails to completely visualize the flow patterns in Hele-Shaw cell experiments. Using a colored solute in water to introduce the density contrasts simultaneously allows the accurate visualization of the solute concentration distribution with high-resolution light transmission measurements (Slim et al., 2013). Contrary to the dense measurements of the concentration distribution, the velocity field that describes the movement of the fluid is experimentally inaccessible.

Optical flow estimation is a classical task in computer vision with the aim of estimating the two-dimensional vector field representing the apparent motion of objects in an image sequence. Typical applications are in autonomous driving (Janai et al., 2017) and action recognition (Simonyan & Zisserman, 2014a). The introduction of supervised deep learning using convolutional neural networks (CNNs) to the field of optical flow estimation in conjunction with training on synthetic data (Fischer et al., 2015) has led to a paradigm shift (Ilg et al., 2017). CNNs with an encoder-decoder architecture to estimate motion showed state-of-the-art results on benchmark data sets, while enabling the estimation in real time.

De Bezenac et al. (2017) applied an adapted encoder-decoder CNN to a related system, the prediction of synthetically generated sea surface temperature data described by convection-diffusion. For this example they showed that the method can learn the underlying processes such that it is competitive with a numerical assimilation method. Zhu and Zabaras (2018) used encoder-decoder CNNs as surrogate models for uncertainty quantification in modeling steady-state single-phase flow in heterogeneous media. With Bayesian treatment of the CNN by adopting the variational inference method of Liu (2017) and Liu and Wang (2016), they showed improved scalability to high-dimensional problems with limited training data. Mo et al. (2019) adopted the network architecture of Zhu and Zabaras (2018) and extended the model as a surrogate for uncertainty quantification of transient multiphase flow in heterogeneous media. Whereas these studies focus on replacing the forward models of related physical systems using an encoder-decoder CNN, in our work we use similar deep learning methods to aim at the estimation of missing system quantities. Generally, these challenges are subject to methods like inverse modeling and data assimilation.

In this study, we explored the information transfer from synthetically generated data, representing the process understanding, to laboratory experiments with missing velocity field data using recent deep learning methods. For the information transfer to be coherent, this requires (i) the physical processes that occur in the laboratory experiment to be represented completely and faithfully in the physical model and therefore in the synthetic data and (ii) the experimental and the synthetic data to have the same, in our case image-like, structure. In a first step we generated a set of synthetic training data through numerical simulation of density-driven active solute transport. Based on FlowNet by Fischer et al. (2015) and FlowNet 2.0 by Ilg et al. (2017) and analogously to de Bezenac et al. (2017), Zhu and Zabaras (2018), and Mo et al. (2019), we trained an encoder-decoder CNN, an adaption of the FlowNet2-s (Ilg et al., 2017), on the synthetic training data in an end-to-end fashion. In the encoder high-level features in the input concentration fields are extracted and transferred to a coarse abstract representation that is refined in the decoder to reconstruct the output velocity fields. The trained CNN was then used to estimate velocity fields from concentration measurements of a synthetic test data set. This test data set was again obtained from numerical simulation, but with the concentration boundary condition being modified to test the generalization of the method. In the next step we applied the CNN to concentration measurements of a Hele-Shaw cell experiment. This way we estimated the velocity fields that are otherwise inaccessible for density-driven flow and assessed the representation of the relevant physical transport processes in the numerical simulation.

With this approach, the CNN implicitly learns, relying purely on synthetic data, the phenomenology of a class of physical processes on a broad parameter spectrum. This incorporated representation of the physical processes can then be directly utilized to estimate missing system quantities on measurements. In this sense

we see this as a complementary approach to inverse modeling and data assimilation of physical processes. These methods are based on accurate models of the processes, where the missing system quantities are reconstructed by calibrating the forward model on the measured data.

2. Materials and Methods

2.1. Physical Representation

2.1.1. Dynamics

Fluid flow in saturated isotropic porous media is governed by Darcy's law

$$\mathbf{u} = -\frac{k}{\mu} [\nabla p - \rho g \mathbf{e}_z] \quad (1)$$

and the continuity equation

$$\nabla \cdot \mathbf{u} = 0 \quad (2)$$

using the Boussinesq approximation. Note that for the special case of a Hele-Shaw cell, the description is reduced to two dimensions with the z direction chosen to coincide with the direction of gravity. Then $\mathbf{u} = (u, w)$ denotes the Darcy velocity, k the permeability, μ the fluid viscosity, p the pressure, and g the acceleration due to gravity. The density of the fluid ρ is considered to only depend on the concentration of the solute C . Solute transport is represented by the convection-dispersion equation

$$\phi \frac{\partial C}{\partial t} = -\mathbf{u} \cdot \nabla C + \phi D \nabla^2 C \quad (3)$$

with porosity ϕ , effective dispersion coefficient D , and Darcy velocity \mathbf{u} , given by equation (1). The coupling between (1) and (3) is described by a linearized relation for the density

$$\rho(C) = \rho_0 + \Delta \rho \tilde{C} \quad (4)$$

with the dimensionless concentration

$$\tilde{C} = [C - C_0] / [C_{\max} - C_0]. \quad (5)$$

This linearization is considered to be valid for small concentration variations within a range from the initial base density ρ_0 at the initial concentration C_0 to the maximum density $\rho_{\max} = \rho_0 + \Delta \rho$ at the maximum concentration C_{\max} .

2.1.2. Dimensionless Formulation

The dimensionless formulation then facilitates the comparison of the numerical and the laboratory experiment. Following the usual procedure (e.g., Riaz et al., 2006; Wooding et al., 1997), we choose to rescale by the depth of the flow domain H as the characteristic length scale L_c . As characteristic velocity U_c we choose the buoyancy velocity of a fluid element with maximum concentration C_{\max} . For the remaining characteristic quantities this choice results in

$$L_c = H, \quad U_c = \frac{\Delta \rho g k}{\mu}, \quad T_c = \frac{L_c}{U_c}, \quad P_c = \frac{\mu U_c L_c}{k}, \quad C_c = C_{\max} - C_0 \quad (6)$$

with characteristic time T_c , characteristic pressure P_c , and characteristic concentration C_c . For the rescaling of equations (1) to (3) we choose

$$\mathbf{u} = U_c \tilde{\mathbf{u}}, \quad \mathbf{x} = L_c \tilde{\mathbf{x}}, \quad t = \phi T_c \tilde{t}, \quad p - \rho_0 g z \mathbf{e}_z = P_c \tilde{p}, \quad C - C_0 = C_c \tilde{C} \quad (7)$$

with $\tilde{\cdot}$ denoting dimensionless quantities. The governing equations take the form

$$\tilde{\mathbf{u}} = -\tilde{\nabla} \tilde{p} + \tilde{C} \mathbf{e}_z \quad (8)$$

$$\tilde{\nabla} \cdot \tilde{\mathbf{u}} = 0 \quad (9)$$

$$\frac{\partial \tilde{C}}{\partial \tilde{t}} = -\tilde{\mathbf{u}} \cdot \tilde{\nabla} \tilde{C} + \frac{1}{Ra} \tilde{\nabla}^2 \tilde{C}, \quad (10)$$

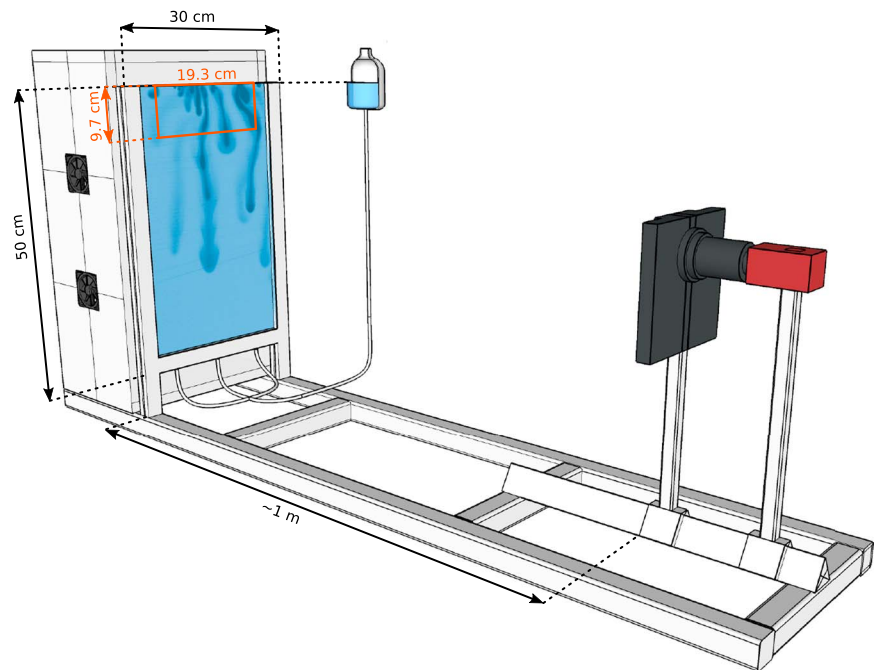


Figure 1. Illustration of the experimental setup: The Hele-Shaw cell (50 × 30 cm) is placed in front of the LED light source. During the evaporation-induced density-driven instability experiments the external fluid reservoir keeps the fluid level inside the Hele-Shaw cell stable. The CCD camera captures the transmitted light. The red rectangle highlights the observation area used for the velocity estimation.

where the Rayleigh number

$$Ra = \frac{\Delta\rho gkH}{\mu\phi D} \quad (11)$$

emerges in the convection-dispersion equation as the scaling parameter between the convective and the dispersive terms.

2.2. Laboratory Experiment

The experimental setup (Figure 1) is composed of a Hele-Shaw cell, an LED light source, an optical band-pass filter, and a CCD camera to observe the solute concentration distribution by employing light transmission measurements.

The Hele-Shaw cell is made from two 8-mm-thick glass plates with the dimensions of 500 × 300 mm. Spacers of 0.4-mm thickness and sealing material along the left, right, and lower edges hold the glass plates apart to create a narrow gap d that serves as the quasi two-dimensional flow domain. In this narrow gap, the fluid flow is described by the same governing equations as for porous media (see section 2.1) with porosity $\phi = 1$. We ensured that the gap width satisfies $d < 10(\phi D/U_c)$, as deduced by Slim et al. (2013) from the results of Fernandez et al. (2002), in order to reduce the influence of possible three-dimensional effects.

Three inlet ports at the lower edge were connected to an external fluid reservoir to keep the fluid level stable during the course of the experiment. The fluid used in the experiment is deionized water with *Brilliant Blue FCF* (BB) as a tracer and solute simultaneously that allows to observe the dynamics as well as to introduce density contrasts depending on the BB concentration. Based on the Beer-Lambert law, we determined the absorption coefficient of BB using the measured light extinction in a spectrometer.

The light source consists of an LED array and a translucent acrylic glass plate as optical diffuser up front to create a homogeneous illumination of the Hele-Shaw cell. The LEDs match the maximum light absorption wavelength of BB (630 nm) in their maximum emission wavelength (625 nm). The optical band-pass filter in front of the CCD camera (2,452 × 2,054 pixels) was chosen accordingly with a transmission band of 632 ± 11 nm.

Table 1
Summary of the Synthetic Data Sets

Characteristics	Training data set	Validation data set	Test data set
Number of image pairs	5,347	269	829
Range of Ra	[2,000, 27,000]	[2,000, 27,000]	[3,750, 13,750]
Resolution	768 pixels \times 384 pixels	768 pixels \times 384 pixels	768 pixels \times 384 pixels
Upper boundary condition	$\tilde{C} = 1$	$\tilde{C} = 1$	$\tilde{C} = 1$ for 230 pixels $< x < 538$ pixels and $\tilde{t} < \frac{5}{6}T_c$; $\tilde{C} = 0$ else

To determine the spatially resolved gap width of the Hele-Shaw cell, we performed calibration measurements using two uniform BB concentration solutions. By measuring the light intensity with the CCD camera the width can be inferred by the Beer-Lambert law. This calibration for the gap width allowed for the spatially resolved determination of the BB concentrations during the experiment.

For the density-driven instability experiment, the flow domain was initially completely filled with uniform BB concentration solution ($C_0 = 86.5 \text{ kg/m}^3$). Through water evaporation at the upper unsealed edge of the Hele-Shaw cell, BB accumulated locally up to $C_{\max} = 154.4 \text{ kg/m}^3$. This set the upper boundary condition for the experiment. The fluid loss due to evaporation was compensated by inflow of BB solution with C_0 through the three inlet ports at the bottom. Note that this increased the total BB amount in the cell over time. The increased BB concentration at the top altered the density ($\Delta\rho = [12.9 \pm 2.8] \times 10^{-3} \text{ kg/m}^3$) of the solution there leading to an unstable layering within the fluid. Subsequent to this the evaporation-induced density-driven instability developed in its full beauty (video available at <https://doi.org/10.11588/data/7NEEKF>). For the laboratory experiment we estimated the Rayleigh number to be $Ra = 6,600 \pm 2,700$, where the large uncertainties are mainly attributed to large uncertainties in the diffusion coefficient D .

2.3. Numerical Experiment

To simulate the dynamics of the problem as formulated by (8)–(10), we used a numerical solver implemented by P. Bastian in DUNE (Bastian et al., 2008a, 2008b; Blatt & Bastian, 2007, 2008; Blatt et al., 2016). We ran the simulations on a two-dimensional rectangular structured 768×384 grid to generate three distinct data sets: (i) the training data set, (ii) the validation data set, and (iii) the test data set each containing the concentration fields and their underlying velocity fields grouped into image pairs of consecutive time steps. Examples of a concentration field and the corresponding velocity field are shown in Figure 3a.

For the training data set the concentration was set to a constant value of $\tilde{C} = 1$ at the upper boundary. The upper and lower boundaries were set to be impermeable to the fluid flow. In the range of $Ra \in [2,000, 27,000]$ we ran 27 individual simulations to generate the training data set. Out of these image pairs we equidistantly excluded 269 image pairs, which will subsequently be used as the validation set, resulting in a training data set of 5,347 image pairs. The test data set was generated in the range of $Ra \in [3,750, 13,750]$ over the course of six simulation runs yielding 829 image pairs. In contrast to the training and validation data sets, here the concentration was set to $\tilde{C} = 1$ only in the middle two fifths of the upper boundary (cf. Figure 4a) and a temporal cutoff was introduced as the concentration was set to $\tilde{C} = 0$ at $\tilde{t} = \frac{5}{6}T_c$. This difference in the boundary conditions was introduced to test the generalization of the velocity field estimation. A summary of the training, validation, and test data set is given in Table 1.

The amount of data available for training is crucial to the performance of a supervised deep learning method. Larger data sets like Flying Chairs (Fischer et al., 2015) (22,872 image pairs for training) and FlyingThings3D (Mayer et al., 2016; over 35,000 image pairs for training) are beneficial over smaller sets like KITTI (Geiger et al., 2013; 194 image pairs for training) and MPI-Sintel (Butler et al., 2012; 1,041 image pairs for training) for optical flow. On the other hand, the generation of larger data sets requires additional computational resources. Our chosen data set size (5,347 image pairs for training) balances the two aspects for this exploratory case.

2.4. Data Preprocessing

To prepare the data of the laboratory experiment for the velocity field estimation, we used the following preprocessing steps. To reduce the image noise, we used a standard median filter with a kernel size of 5×5 pixels. A median image filter, in comparison to a Gaussian image filter, keeps the blurring minimal and therefore reduces the introduced error with respect to the dispersive transport process. Additionally, we used

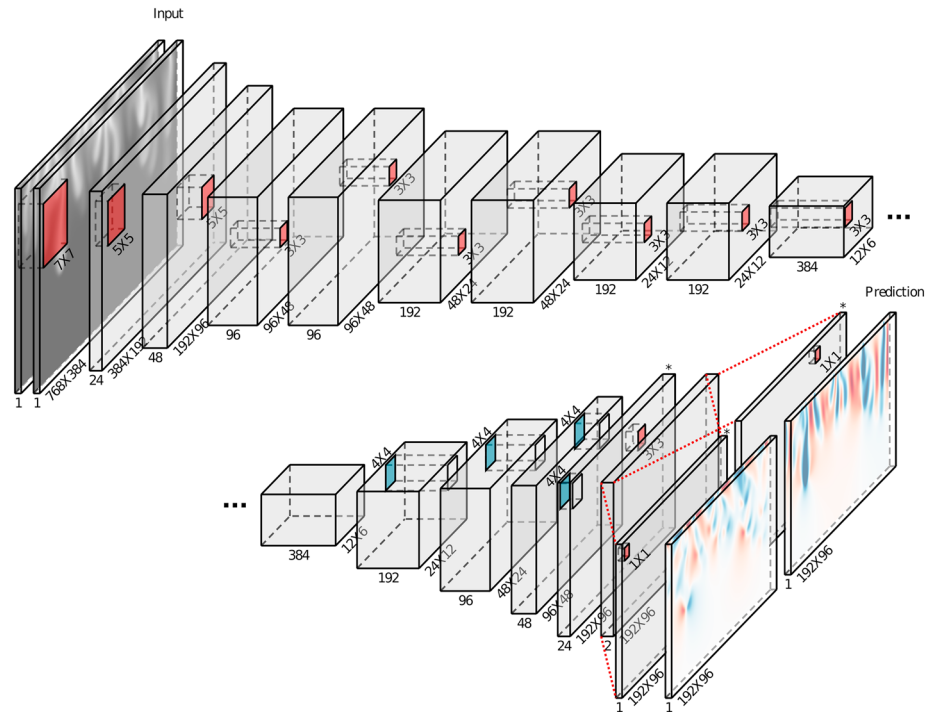


Figure 2. Architecture of the convolutional neural network (details in Table A1): Two subsequent concentration fields are fed into the network. Feature maps are depicted as light gray boxes. Convolution layers are depicted as boxes with red faces indicating their mapping from the respective feature map to the next. Transposed convolution layers are depicted as boxes with blue faces indicating their mapping from the previous to the respective feature map. Asterisks indicate convolution layers without leaky rectified linear unit activation function. The dotted red lines indicate slicing of the feature map into x and y components of the predicted velocity field. Bilinear interpolation as postprocessing of the predicted velocity field is used to recover the full input resolution (not depicted).

a threshold filter slightly above the base concentration of $C_0 = 86.5 \text{ kg/m}^3$, setting all concentration values beneath $C = 90.0 \text{ kg/m}^3$ to 0 to extract the density fingers only.

To enable the direct comparison to the numerical experiment, we rescaled the measured concentration fields according to equation (5). As we are only interested in the early to intermediate development of the system, we chose to restrict our observations to the upper 9.7 cm of the Hele-Shaw cell with an observation window width of 19.3 cm (cf. Figure 1). With this we received the resolution of 768×384 pixels for the concentration fields.

2.5. Velocity Field Estimation

The architecture we used for the CNN is an encoder-decoder very similar to the FlowNet2-s architecture as proposed in Ilg et al. (2017). An illustration of our CNN is given in Figure 2, and a detailed description of the architecture and the training scheme is given in Appendix A. Our architecture differs in the following aspects: (i) We omitted the skip connections, as we did not encounter high-frequency features in our data. (ii) For data augmentation we solely accounted for image noise. (iii) We introduced a convolution layer with a single 1×1 convolution filter for the individual velocity components right before the loss to enhance the networks adaptability of the velocities specific to our data. Additionally, internal parameters to scale the data range were adapted to allow for good information propagation through the network.

As input the CNN takes two subsequent concentration fields, concatenated to produce the input image, and outputs a velocity field (cf. Figure 2). For the training of the CNN we used the modified version of the Caffe framework (Jia et al., 2014) as provided by Ilg et al. (2017) on a single Nvidia GTX 1080 Ti graphics processing unit. The trained CNN was then deployed to estimate the velocity field that accounts for the solute transport between the two input concentration fields. Since the encoder and the decoder of the CNN are asymmetric, the resolution of the data from input to output is coarsened by a factor of 4. To restore the original resolution, we used bilinear interpolation of the predicted velocity fields as a postprocessing step.

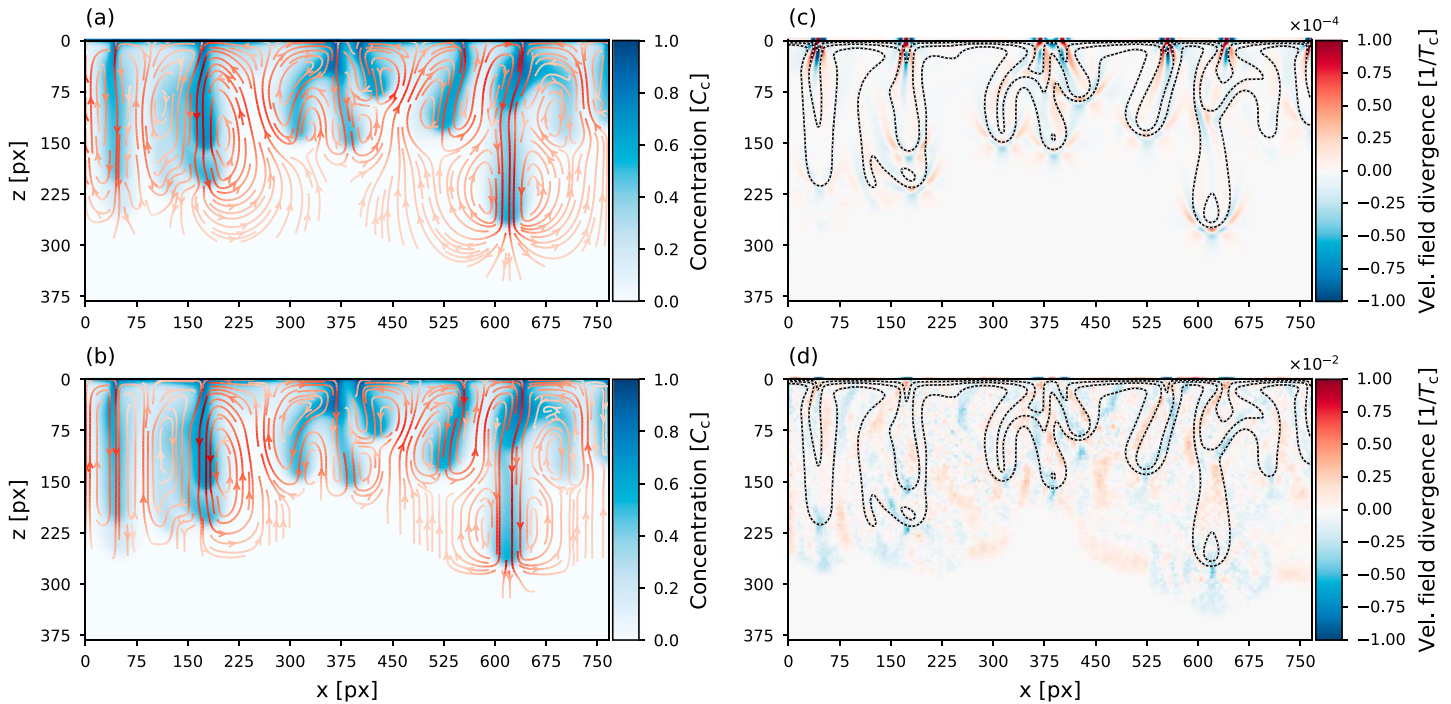


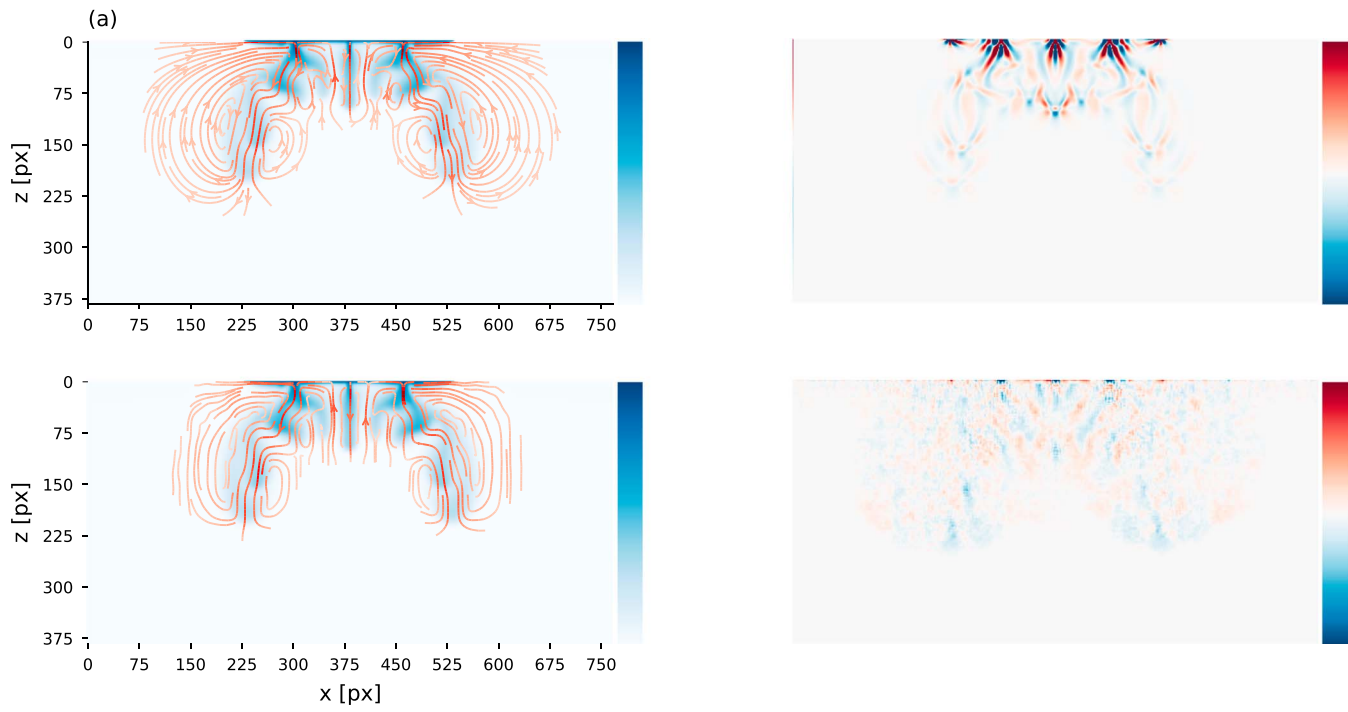
Figure 3. Estimated velocity field on the synthetic validation data set example: truth (a) and estimation (b) of the velocity field shown as streamlines (red color intensity indicates absolute velocity) on top of the color-coded prior concentration field. Velocity field divergence of the true (c) and estimated (d) velocity field. Concentration isolines are given at levels $\bar{C} = (0.25, 0.5, 0.75)$. Note that for better visibility of low values the scales for the divergence differ by 2 orders of magnitude, while the actual deviation between maxima and minima is lower.

2.6. Concentration Field Propagation

When estimating the velocity fields with the trained CNN, ground truth is only available for the numerical experiment but not for the laboratory measurements. To be able to evaluate the performance of the estimation, we propagated the concentration fields forward in time using the estimated velocity fields. The resulting estimated concentration fields were then compared to the concentration fields of the corresponding time step as a basis for the performance evaluation of the velocity field estimation. The propagation was performed through two subsequent processing steps: (i) We used warping, the bilinear interpolation of the propagated concentration values (Ilg et al., 2017), to account for the convective transport process, and (ii) we used a standard Gaussian image filter to account for the dispersive transport process.

As the training of the CNN was performed based on dimensionless simulations, the estimations contain dimensionless velocities $\bar{\mathbf{u}}$. Therefore, we needed to rescale these velocities with the dimensionless time step $\bar{d}\bar{t}$ to get the displacement for the warping method. For the synthetic data this is given by the time step of the numerical simulation $\bar{d}\bar{t} = 0.02$. With the known dispersion coefficient of $D = 1/Ra$ in the numerical simulations the standard deviation for the Gaussian image filter is given by $\sigma = \sqrt{2\bar{d}\bar{t}/Ra}$.

For the laboratory experiment we lack the accurate knowledge of the dimensionless time step $\bar{d}\bar{t}$ and the standard deviation for the Gaussian blurring σ , because of uncertainties in the experimental determination of $T_c = \mu H / \Delta \rho g k$ and D . For the estimated time step we received $\bar{d}\bar{t} = dt / T_c = 0.039$. Optimizing the values of time step and standard deviation for the warping and Gaussian filtering of the measured data such that the mean squared difference between the propagated and the corresponding measured concentration field is minimized resulted in $\bar{d}\bar{t} = 0.0438$ and $\sigma = 2.6$ pixels. When comparing the value of the time steps, we encountered a deviation of 11% between the estimated value and the value from optimization. This deviation lies well in the uncertainties of the experimentally determined value. We chose the optimized time step since the estimation from the measurements is expected to be biased.



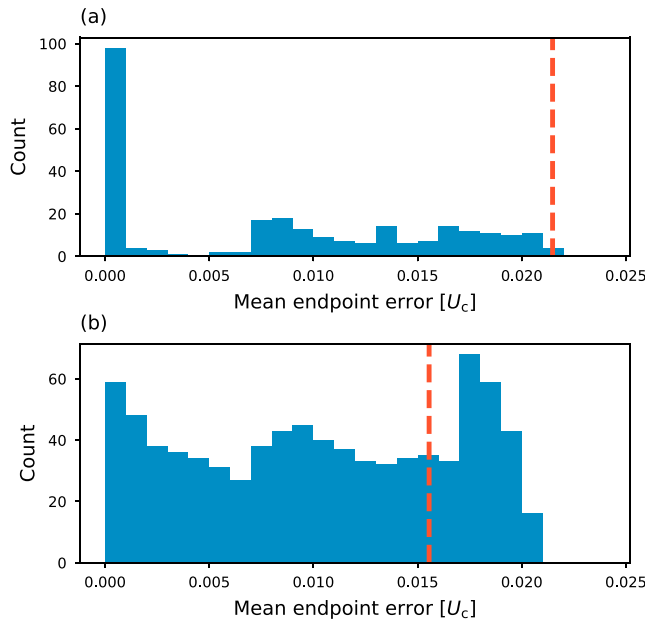


Figure 5. Error distributions of the mean endpoint error for the validation data set (a) and the test data set (b). Mean endpoint error values of the synthetic data set examples, shown in Figures 3 and 4, respectively, are indicated by the red dashed lines.

For the training data set the MEPE remains below 0.022 for all the image pairs suggesting a good agreement between the estimation and the truth. Image pairs with very little movement result in low values beneath 0.001. In fact, the majority of the data count in this range originated from image pairs during the purely diffusive regime. For the image pairs after the instability onset, the majority is contained within the MEPE range from 0.007 to 0.022.

The distribution of the MEPE over the test data set (Figure 5b) is similar to the distribution over the validation data set with the maximum MEPE being 0.021. Note that for the test data set the flow is being contained in smaller regions, given the nonuniform boundary condition (see section 2.3). In contrast to the MEPE distribution over the validation data set, the portion of image pairs with mean endpoint error below 0.001 is lower. This is due to the composition of the test data set. With the nonuniform boundary condition we introduced an earlier onset of the instability there. In the range between 0.001 and 0.021 we observe a broad distribution of the MEPE. Overall, the distribution exhibits that the estimation performs well on the test data set, showing the robustness of the method toward the spatial and temporal variability introduced by the different upper boundary conditions.

Limitations of the velocity field estimation with the implementation of the CNN we used are in the accurate estimation of the absolute values of the flow velocities. This is especially true in regions with large velocities. To get an assessment of the spatial distribution of the errors, we calculated the normalized velocity error as local error measure. Additionally, we give the angular error. The results for the validation data example and the test data example are shown in Figures 6 and 7, respectively, whereas the respective MEPE is indicated in Figures 5a and 5b by the dashed red lines. The normalized velocity error is the difference between the absolute estimated velocity field and the absolute true velocity field normalized by the maximum absolute value of the true velocity field $[|\hat{\mathbf{u}}^{\text{est}}| - |\mathbf{u}^{\text{true}}|] / \max(|\mathbf{u}^{\text{true}}|)$ and yields structurally similar results to the L^2 error often used in the field of deep learning. The angular error shows the angular difference between the estimated velocity direction and the true velocity direction for each pixel and is given in degrees. To show the contributions of the individual velocity components to the normalized velocity error, we additionally present the normalized x and z velocity differences $[\hat{u}_i^{\text{est}} - u_i^{\text{true}}] / \max(|\mathbf{u}^{\text{true}}|)$ with $i = x, z$ in Figures 6c, 6d, 7c, and 7d, respectively.

Figures 6a and 7a show that the direction of the velocity is generally estimated accurately, where information is given by the concentration. The CNN is even capable of estimating the velocity direction accurately in larger regions surrounding the density fingers. Areas where the estimation of the velocity direction fails are located directly at the flanks of density fingers, where the velocity is typically small and deviates from

Figures 3c, 3d, 4c, and 4d present the corresponding velocity field divergence values for the true and the estimated velocity fields for the validation data set and the test data set examples. For the true velocity fields from numerical simulation we find divergences in the range $[-9.3, 8.6] \cdot 10^{-3}$ and divergences in the range $[-30.0, 9.0] \cdot 10^{-3}$ for the estimated velocity fields. The larger range of divergences for the estimated velocity fields arises from the imperfect estimation of the CNN and is related to the velocity errors occurring per characteristic time period T_c . Still, the encountered values are at least 2 orders of magnitude smaller than the encountered typical velocities of $|\hat{\mathbf{u}}| \approx 3.5$.

As a quantitative error measure we took the mean endpoint error (MEPE = $\sum_{j=1}^N \sqrt{[\hat{u}_j^{\text{est}} - u_j^{\text{true}}]^2 + [\hat{v}_j^{\text{est}} - v_j^{\text{true}}]^2} / N$ with N being the number of pixels in the images) between the estimated and the true velocity field. The distributions of this error measure over the validation data set and the test data set are shown in Figures 5a and 5b, respectively. The MEPE is given in the units of the characteristic velocity U_c and is a measure of the error in velocity averaged over all pixels of the respective image pair. Typical absolute velocity values encountered after the onset of the instability are in the range of $|\hat{\mathbf{u}}| \approx 0.5 U_c \dots 0.8 U_c$. Note that the MEPE is expected to result in low values for image pairs with very little to no movement and hence for the very early development of the instability.

For the training data set the MEPE remains below 0.022 for all the image pairs suggesting a good agreement between the estimation and the truth. Image pairs with very little movement result in low values beneath 0.001. In fact, the majority of the data count in this range originated from image

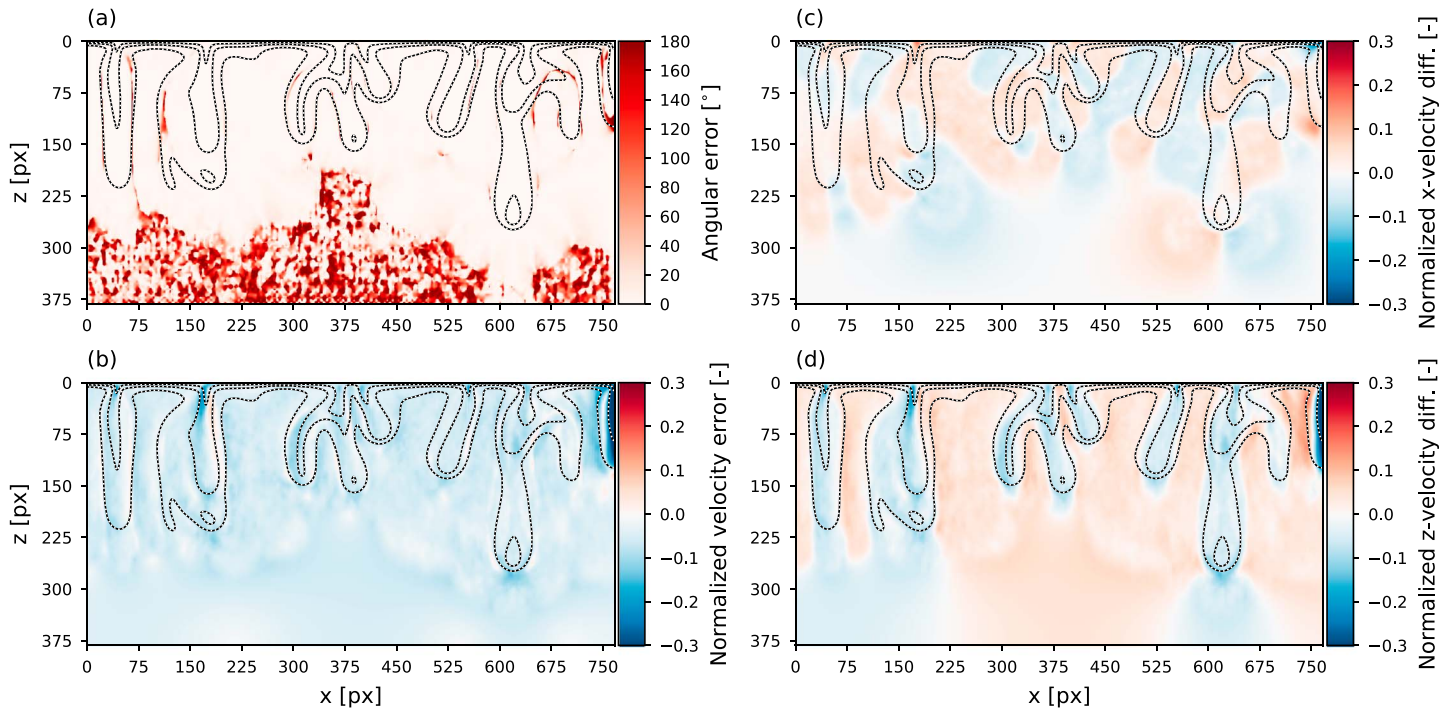


Figure 6. Error measures for the validation data set example: angular error (a), normalized velocity error (b; negative for $|\bar{\mathbf{u}}^{\text{est}}| < |\bar{\mathbf{u}}^{\text{true}}|$), normalized x velocity difference (c), and normalized z velocity difference (d). Concentration isolines are given for the levels $\tilde{C} = (0.25, 0.5, 0.75)$.

being parallel to the orientation of the density fingers. Additionally, the concentration values are low. The combination of these two factors results in low dynamics impeding the velocity field estimation there. Also, boundary effects become apparent in the validation data example (Figure 6a) at the rightmost density finger. There the flow points downward directly at the right boundary causing the shape of the finger to be structurally different as it appears to be cut in half. In the regions further away from the density fingers, the velocity is close to 0, the angle correspondingly ill defined.

Figures 6b and 7b show that the method predominantly underestimates the velocity values. Apart from the boundary effects on the rightmost density finger in the validation data set example (Figure 6b), large portions of the region where $\tilde{C} > 0$ exhibit a normalized velocity error roughly in the range between -13% and 6% also with a strong tendency to underestimate the velocity. Locally, the error can exceed these values. For instance, at the seeding point of the second density finger from the left the normalized velocity error ranges down to -23% . Also, the velocities are underestimated in the regions with upward flow between the density fingers. There the normalized velocity error roughly ranges from -7% to -4% . Figure 7b shows that the underestimation is generally less pronounced for the test data set example. This is mainly due to the predominantly lower velocities (cf. Figures 3a and 4a). However, the span of the normalized velocity error is similar to the validation data set example with a range from -20% to 6% .

To be able to compare the quality of the results on synthetic and real data (where the ground truth is inaccessible), we used concentration field propagation by subsequent warping and Gaussian image filtering (see section 2.6). We investigated the concentration field propagation on the synthetic data to obtain a reference. The upper boundary condition for the concentration was set to $\tilde{C} = 1$ resembling the boundary condition in the numerical simulation. We chose one distinct concentration field with the propagation time $\tilde{t}_p = 0$ as the starting point. The concentration field was then iteratively propagated by the propagation time step $d\tilde{t} = 0.02$, since the true concentration field was present in this temporal resolution. For the chosen concentration field, the given propagation time step resulted in $\sigma = 2.43$ pixels for the Gaussian blurring.

Example results of the concentration field propagation for propagation times $\tilde{t}_p = 0.04$, $\tilde{t}_p = 0.08$, and $\tilde{t}_p = 0.22$ are presented in Figure B1. The true concentration fields are shown in Figures B1a–B1c, the concentrations field propagated with the true velocity fields are shown in Figures B1d–B1f, and the concentration fields propagated with the estimated velocity fields are shown in Figures B1g–B1i.

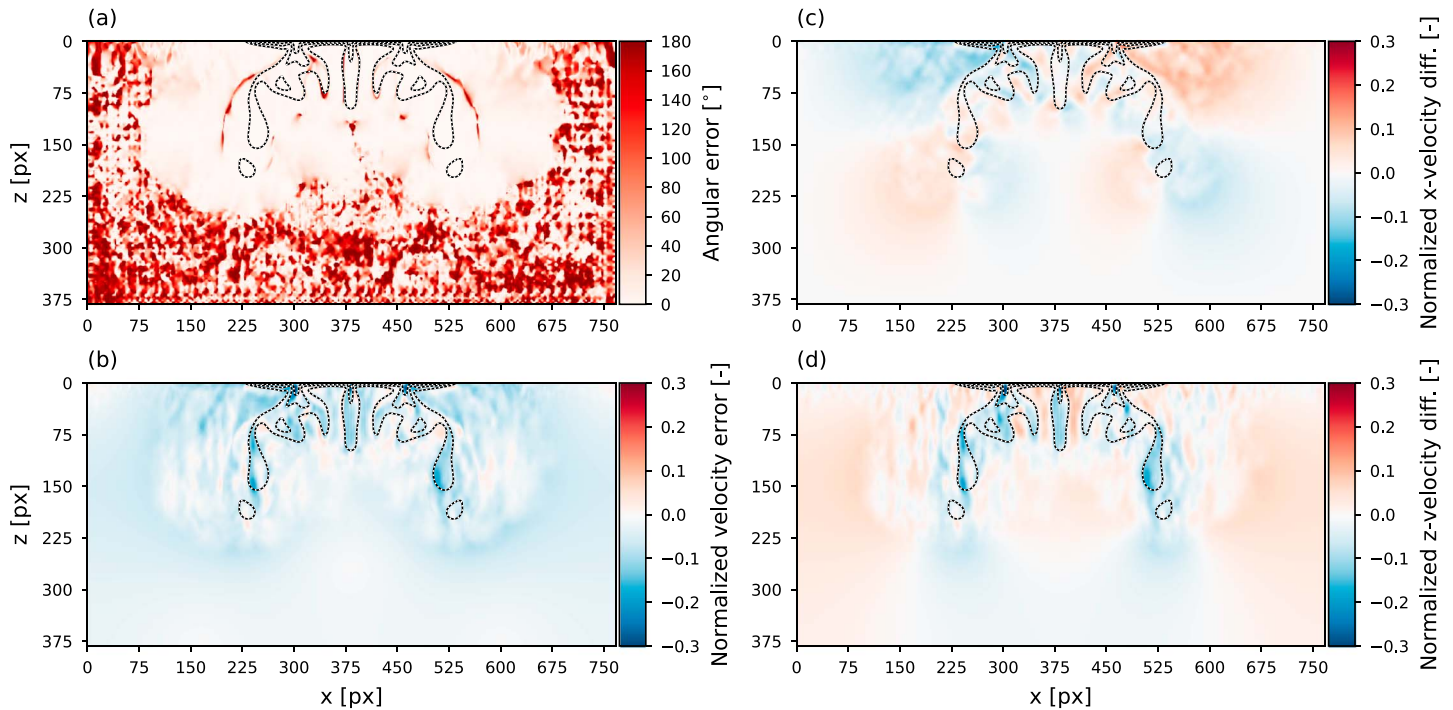


Figure 7. Error measures for the test data set example. Same representation as in Figure 6.

A detailed look at the deviations between the concentration fields propagated with the estimated velocity fields and the true concentration fields is presented in Figures 8a–8c as the normalized concentration errors ($[(\tilde{C}^{\text{est}} - \tilde{C}^{\text{true}}) / \max(\tilde{C}^{\text{true}})]$) for the same propagation times $\tilde{t}_p = 0.04$, $\tilde{t}_p = 0.08$, and $\tilde{t}_p = 0.22$, respectively. At $\tilde{t}_p = 0.04$ the deviations in the middle region, where the velocities are relatively small, roughly range from -1.5% up to 1.5% . In regions with higher velocities on the left the concentration field deviates stronger. Especially at the tip of the furthest developed density finger (at $x = 50$) the normalized concentration error is -3% . This is consistent with the already observed underestimation of the absolute velocities (cf. Figures 6b and 7b). The solute there is not transported as far as in the true concentration field. For the subsequent propagation times the estimated and the true concentration fields diverge even further, also as expected due to the underestimation of the velocity fields.

3.2. Laboratory Experiment

For the laboratory experiment the velocity fields were estimated with the trained CNN, and bilinear interpolation was used to recover the original data resolution. Figure 9 shows the results on an exemplary concentration field. The quality of the velocity field estimation is consistent with the estimation for the numerical experiment (cf. Figures 9a, 3b, and 4b). Predominantly the estimation of the flow structures seems to be reliable. Also, the resulting divergence of the estimated velocity field (Figure 9b) is consistent with the synthetic case (cf. Figure 3d). Nevertheless, the ground truth for the laboratory experiment is inaccessible for direct performance evaluation. To assess the performance on the real data, we propagated the measured concentration fields forward in time using the estimated velocity fields (see section 2.6). The propagated concentration fields were then compared to the respective measured concentration fields.

As described in section 2.6 we chose to optimize for the propagation time step and the Gaussian image filter standard deviation, although we are aware that this alleviates the expected underestimation of the velocity field with the CNN. On the other hand, the physically rescaled value for $d\tilde{t}$ introduces estimation biases that cannot be assessed. Our choice results in $d\tilde{t} = 0.0438$ and $\sigma = 2.6$ pixels. Additionally, in the laboratory experiment we do not know the upper boundary condition for the concentration. Therefore, we took the upper 10 pixel lines of the measured concentration field and set these pixel lines as the upper boundary condition before every warping step. Using these parameters for the concentration field propagation, we received the estimated concentration fields as shown in Figures 10d–10f together with the respective measured concentration fields in Figures 10a–10c. The propagation times are $\tilde{t}_p = 0.0438$, $\tilde{t}_p = 0.0876$, and

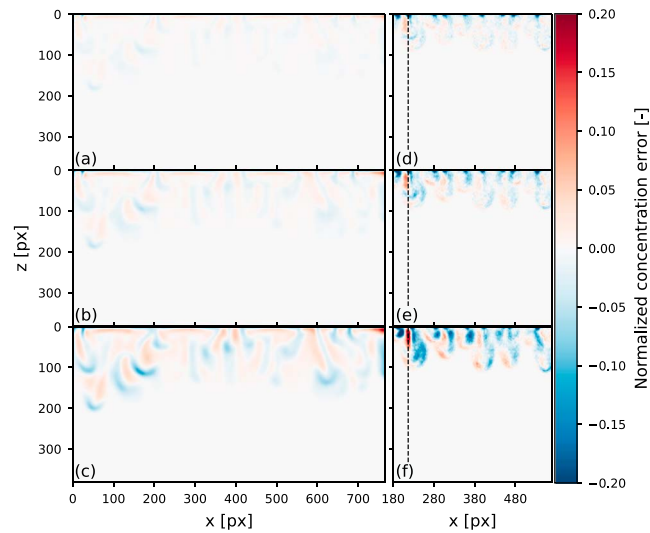


Figure 8. Normalized concentration errors for propagated concentration fields. Numerical experiment: (a) at $\tilde{t}_p = 0.04$, (b) at $\tilde{t}_p = 0.08$, and (c) at $\tilde{t}_p = 0.22$. Laboratory experiment: (d) at $\tilde{d}t = 0.0438$, (e) at $\tilde{t}_p = 0.0876$, and (f) at $\tilde{t}_p = 0.219$. Normalized concentration errors are chosen to be negative for $\tilde{C}^{est} < \tilde{C}^{true}$. Black dashed lines indicate the initial position of the respective density finger seeding point (cf. red dashed line in Figure 10). For the laboratory experiment only the central region is presented (cf. blue dotted lines in Figure 10).

$\tilde{t}_p = 0.219$, respectively. These propagation times were chosen to closely match the propagation times for the numerical experiment as shown in Figures B1 and 8a–8c. To the eye the concentration fields at $\tilde{t}_p = 0.0438$ do agree in general, although slight deviations at the leftmost and rightmost density fingers already become apparent. At $\tilde{t}_p = 0.0876$ and $\tilde{t}_p = 0.219$ these deviations become even larger until the concentration fields do not agree very well anymore. In the center region the agreement in the shape is still very good despite the appearance of the slightly pointier finger tips as already seen in Figure B1i for the synthetic data. We assume that the outermost fingers in the laboratory experiment are additionally affected by boundary effects. Therefore, we chose to limit the evaluation of the normalized concentration errors to the region between $x = 180$ pixels and $x = 570$ pixels as indicated in Figure 10 by the blue dotted lines.

Figures 8d–8f show the normalized concentration errors at $\tilde{t}_p = 0.0438$, $\tilde{t}_p = 0.0876$, and $\tilde{t}_p = 0.219$, respectively. At $\tilde{t}_p = 0.0438$ the normalized concentration error is generally low with values in the larger portion roughly ranging from -10% to 10% . For the longer propagation times the errors increase, as the estimated concentration field diverges more and more from the measured one. Compared to the normalized concentration errors of the numerical experiment (Figures 8a–8c), the errors are typically larger for the laboratory experiment. Nevertheless, in regions with higher velocities, as seen in the region with $x < 200$ pixels for the numerical experiment, errors can reach similar values in both cases for all the presented propagation times. The qualitative structure of the errors differs as well. For the numerical experiment the concentrations are

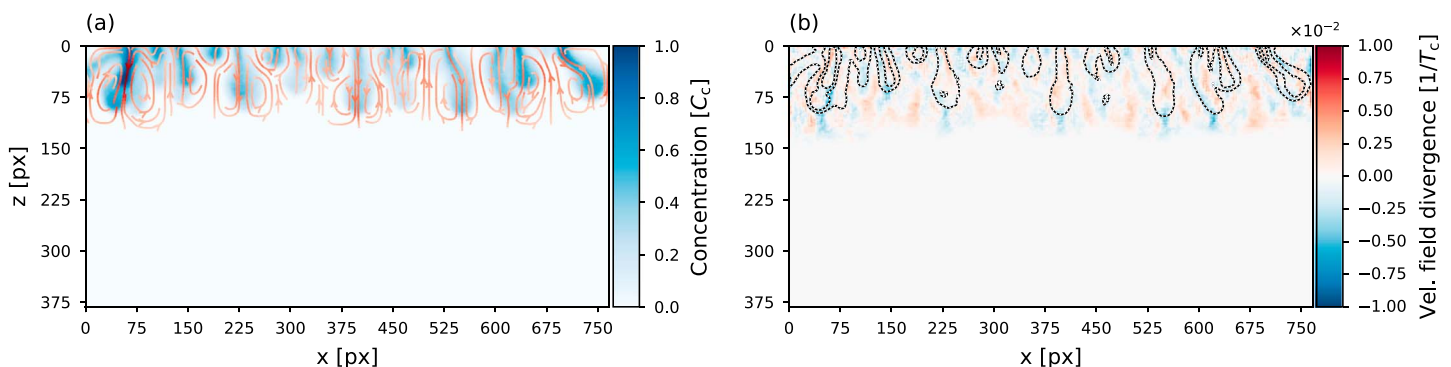


Figure 9. Estimated velocity field on the laboratory experiment (a) shown as streamlines (red color intensity indicates absolute velocity) on top of the color-coded prior concentration field. Velocity field divergence of the estimated velocity field (b). Concentration isolines are given at levels $\tilde{C} = (0.25, 0.5, 0.75)$.

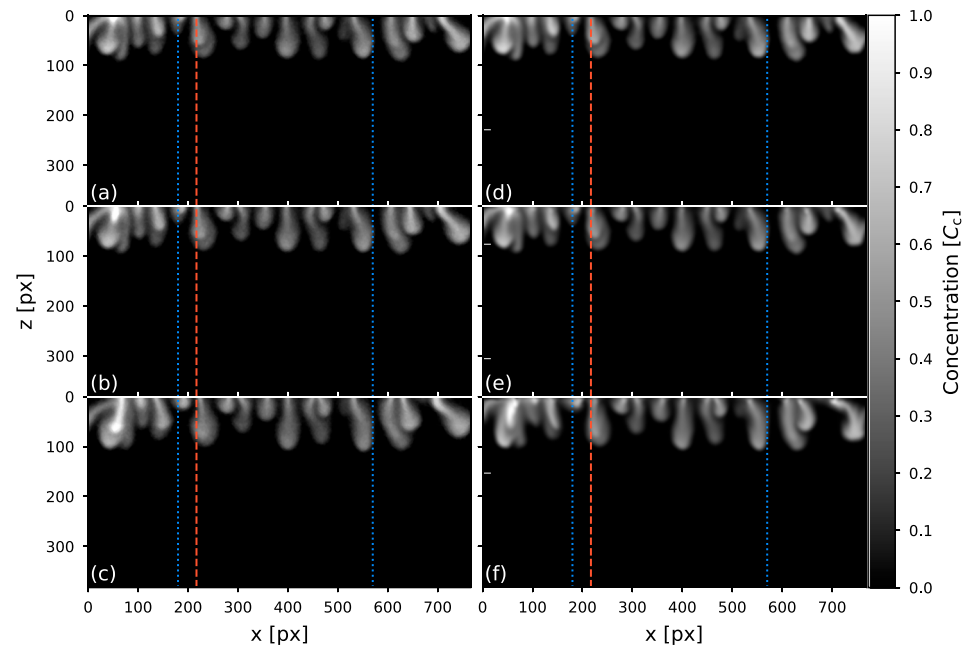


Figure 10. Concentration field propagation on the laboratory experiment: true concentration fields (a–c) and concentration fields propagated with estimated velocity fields (d–f) at propagation times $\bar{t}_p = 0.0438$ (top row), $\bar{t}_p = 0.0876$ (center row), and $\bar{t}_p = 0.219$ (bottom row). The red dashed line marks the initial position of the respective density finger seeding point. The blue dotted lines indicate the region of concentration error evaluation presented in Figures 8d–8f.

underestimated in front of the density finger tips, whereas for the laboratory experiment the concentrations are underestimated on the right flank of the fingers while being overestimated on left flank. This shows that for the numerical experiment the downward flow is too slow whereas for the laboratory experiment the estimation is unable to predict a general rightward flow. As the rightward drift there seems to affect all the fingers quite uniformly we attribute this to convection currents in the Hele-Shaw cell independent of the density-driven flow. Most prominently, the convection is seen in Figures 10a–10c where not only the seeding point of the density finger, initially located at the red dashed line, but the complete finger shifts to the right. The corresponding position in Figures 8d–8f is indicated by the black dashed line. This kind of global convection is not represented in the training data set; hence, the method cannot estimate the velocity fields in this respect.

4. Summary and Conclusion

Based on the work on optical flow estimation using deep learning methods (Fischer et al., 2015; Ilg et al., 2017), we presented a new approach for the estimation of velocity fields for density-driven instabilities. We used numerical simulations to generate a synthetic training data set to train a CNN, which was applied to estimate velocity fields on both synthetic and real-world data without the explicit knowledge of boundary conditions. With this we assessed the consistency of the incorporated physical processes and the information transfer from synthetic to real world data.

Application of the method to synthetic data generally resulted in the reliable estimation of the velocity field directions, showing the structurally correct incorporation of the physical processes in the method. Even the flow directions in regions between density fingers, where no information is given by moving solute, were estimated correctly. So far this was not possible with the application of classical optical flow estimation methods to density-driven instabilities. However, systematic deviations were found in the estimation of the absolute flow velocities, which the method tended to underestimate. This indicates a quantitatively incorrect incorporation of the processes. We mainly attribute this to the specific characteristics of the density-driven instabilities. Unlike in the data usually used in optical flow problems, we only encounter smooth concentration gradients impeding the quantitative correct detection of fluid motion between two subsequent time steps. To resolve the biases, a better adaptation of the deep learning method to the characteristics of the

physical problem is necessary. First adaptations have already been proposed, for instance, loss regularization based on the differential equations for convection-dispersion problems (de Bezenac et al., 2017).

The introduction of spatial and temporal nonuniform concentration boundary conditions in the synthetic test data set showed the potential of the approach. The results agree with the results on the synthetic validation data with uniform boundary condition, demonstrating the robustness of the method toward the introduced variability. This indicates that indeed the underlying physical process was structurally incorporated reasonably well into the CNN, which then can be applied to similar situations. Still, issues with flow boundary conditions were suspected for the laboratory experiment's analysis.

By applying the velocity field estimation to real world data, we transferred information learned on the numerical experiment to the laboratory experiment and assessed the limitations of the method. We investigated the results through propagating the concentration fields with warping and Gaussian image filtering. This showed that the method yielded worse results on real world data than on the synthetic reference. Still, in regions with typically higher velocities, the errors found can be similar. In the laboratory experiment we encountered large-scale convection currents resulting in sideward drift of entire density fingers. This was not covered by the simulations, as the numerical experiment accounted for density-driven flow only. As a consequence, the introduced lateral flow could not be estimated accurately revealing relevant transport processes to be not represented in our case. Nevertheless, we were able to estimate the previously inaccessible main velocity field characteristics for the laboratory experiment.

Given the limitations described above, we showed that the information transfer from synthetic data to real world data with recent deep learning methods can be possible. Still, it is crucial to better adapt the methods to physical problems. In our exploration we were able to show that the representation of density-driven flow can be incorporated into the method, allowing the application to experiments with variability in the concentration boundary conditions. Further, we were able to identify physical processes in the laboratory experiment that were missing in the numerical experiment.

Appendix A: Network Architecture

Our adaptation of the FlowNet2-s (Ilg et al., 2017) consists of an encoder of 10 convolution layers followed by a decoder of 4 transposed convolution layers, with a leaky rectified linear units (leaky ReLU) activation function (Maas et al., 2013) with negative slope of 0.1 following the individual layers. Additional convolution layers were used to predict the estimated velocity fields. Detailed information of the network hyperparameters are summarized in Table A1, and a schematic of the architecture is shown in Figure 2.

At the input of the network we added Gaussian noise with standard deviation uniformly sampled from $[0, 0.04]$ as data augmentation to the concentration fields. As illustrated in Figure 2, each of the first three convolution layers reduces the resolution of the feature maps by a factor of 2. In the following convolution layers, only every second layer reduces the resolution to increase the depth of the network while keeping the resolution at the bottle neck reasonably fine. Including more layers and therefore increasing the nonlinearity with the according leaky ReLU layers is beneficial as it makes the decision function more discriminative (Simonyan & Zisserman, 2014b). In the decoder each of the four transposed convolution layers increases the resolution by a factor of 2. Following the decoder a convolution layer without leaky ReLU layer predicts the velocity estimation and the velocity is split in its x and z components. We introduced another convolution layer (without leaky ReLU layer) with a single 1×1 convolution for each velocity component. This enables the CNN to adapt a scaling of the velocity vector components and leads, in our case, to better estimation of the absolute velocities (comparison not shown).

The coarsening in the encoder effectively reduces the resolution by a factor of 64, while the decoder refines the resolution by a factor of 16, resulting in a total resolution decrease by a factor of 4. We used bilinear interpolation as a postprocessing step on the estimated velocity fields to recover the original resolution of the input concentration fields during deployment of the CNN. Further upsampling using additional transposed convolution layers in the decoder does not necessarily produce significantly better results (Fischer et al., 2015), which is also true in our case. During training we did sample down the true velocity field to match the resolution of the estimated velocity field before the loss was calculated. As training loss we calculated

Table A1
Network and Training Hyperparameters

Layer	N_{filter} , kernel size, stride, pad	Resolution ^a
Network architecture		
Encoder		
Input	—	768 × 384
conv1 ^b	24, 7 × 7, 2, 3	384 × 192
conv2 ^b	48, 5 × 5, 2, 2	192 × 96
conv3 ^b	96, 5 × 5, 2, 2	96 × 48
conv3_1 ^b	96, 3 × 3, 1, 1	96 × 48
conv4 ^b	192, 3 × 3, 2, 1	48 × 24
conv4_1 ^b	192, 3 × 3, 1, 1	48 × 24
conv5 ^b	192, 3 × 3, 2, 1	24 × 12
conv5_1 ^b	192, 3 × 3, 1, 1	24 × 12
conv6 ^b	384, 3 × 3, 2, 1	12 × 6
conv6_1 ^b	384, 3 × 3, 1, 1	12 × 6
Decoder		
convT5 ^b	192, 4 × 4, 2, 1	24 × 12
convT4 ^b	96, 4 × 4, 2, 1	48 × 24
convT3 ^b	48, 4 × 4, 2, 1	96 × 48
convT2 ^b	24, 4 × 4, 2, 1	192 × 96
Prediction		
Predict_conv	2, 3 × 3, 1, 1	192 × 96
Slice	—	192 × 96; 192 × 96
Scale_X_conv; Scale_Z_conv	1, 1 × 1, 1, 0; 1, 1 × 1, 1, 0	192 × 96; 192 × 96
Training hyperparameters		
Optimization algorithm	Adam ^c	$\beta_1 = 0.9, \beta_2 = 0.999$
Learning rate	10^{-5}	
Batch size	24	
Training epochs	540	
Weight decay	$4 \cdot 10^{-4}$	

^aRefers to the resolution of the feature map after the respective network layer. ^bLayer is followed by a leaky ReLU activation function with negative slope of 0.1. ^cKingma and Ba (2014).

the L^2 regularized sum of squares error of the velocity field components:

$$J(\mathbf{w}) = \sum_{j=1}^{N^{\text{est}}} \left[\left[\tilde{u}_j^{\text{est}} - \tilde{u}_j^{\text{true}} \right]^2 + \left[\tilde{w}_j^{\text{est}} - \tilde{w}_j^{\text{true}} \right]^2 \right] + \frac{\alpha}{2} \mathbf{w}^T \mathbf{w} \quad (\text{A1})$$

with network weights \mathbf{w} , weight decay $\alpha = 4 \cdot 10^{-4}$, and the number of pixels N^{est} of the estimated, coarse velocity fields. For the training scheme we used a batch size of 24 and a learning rate of 10^{-5} over 540 training epochs. Following Fischer et al. (2015), we used the Adam optimization algorithm (Kingma & Ba, 2014) with $\beta_1 = 0.9$ and $\beta_2 = 0.999$ for the optimization during the training.

Appendix B: Concentration Field Propagation: Synthetic Reference

Figures B1a–B1i present exemplary results of the concentration field propagation on synthetic data. Figures B1a–B1c show the true concentration fields at the propagation times $\tilde{t}_p = 0.04$, $\tilde{t}_p = 0.08$, and $\tilde{t}_p = 0.22$, respectively. Note that for all Figures in the same row the propagation time step is the same. In Figures B1d–B1f the concentration fields are warped with the true synthetic velocity fields and then blurred with the Gaussian filter. The upper boundary condition for the concentration is set to be $\tilde{C} = 1$ resembling

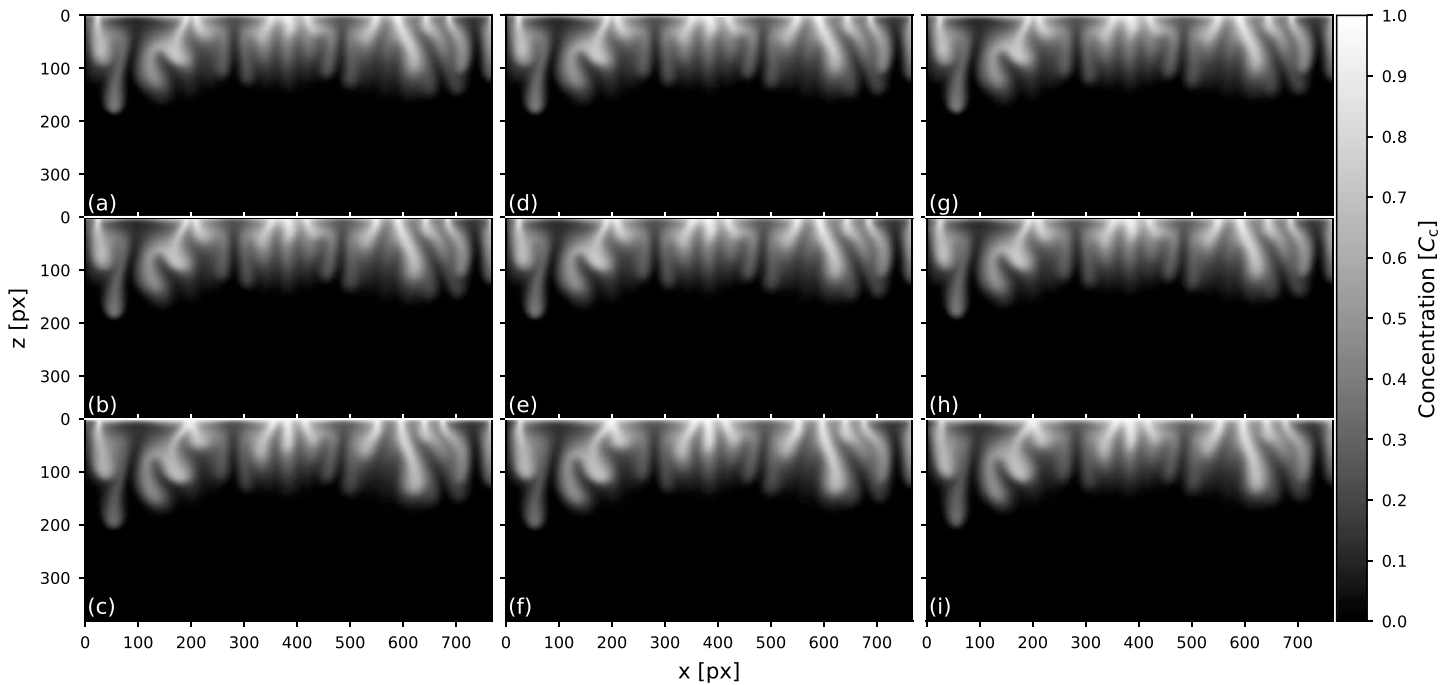


Figure B1. Concentration field propagation by $\tilde{t}_p = 0.04$ (top row), 0.08 (center row), and 0.22 (bottom row) on the numerical experiment: true concentration fields (a–c), concentration fields propagated with true velocity fields (d–f), and concentration fields propagated with estimated velocity fields (g–i).

the boundary condition in the numerical simulation. The resulting concentration fields for warping with the true velocity fields mainly match the true concentration fields very accurate. However, investigation of the normalized concentration error locally reveals large deviations of up to 50% (comparison not shown) confined in the upper left corner. This indicates issues due to warping where pixel values with velocities pointing outward of the domain are set to 0. Focusing on the interior of the domain and excluding this small region results in a maximum absolute normalized concentration error of 4.7% at $d\tilde{t} = 0.22$. This sets the lower boundary for the relative errors we can expect from the estimated velocity fields after this time.

Figures B1g–B1i show the results of the concentration fields with warping by the estimated velocity field. Also here, the general shape of the concentration distribution is reproduced very well. At propagation times $\tilde{t}_p = 0.04$ and $\tilde{t}_p = 0.08$ barely any differences are noticeable, whereas for propagation time $\tilde{t}_p = 0.22$ first deviations become apparent. For instance, the finger tip at $(50, 200)$ differs in shape as it becomes pointier through the warping with the estimated velocity field. Still, the overall flow characteristics are described very well as it is seen for the coalescence of three density fingers at around $x = 400$.

Acronyms

CNN Convolutional neural network
DUNE Distributed and Unified Numerics Environment.
MEPE Mean endpoint error.

Notation

C solute concentration (kg/m^3).
 \tilde{C} dimensionless concentration.
 C_0 initial solute concentration (kg/m^3).
 C_c characteristic concentration (kg/m^3).
 C_{max} maximum solute concentration (kg/m^3).
 D effective dispersion coefficient (m^2/s).
 $d\tilde{t}$ dimensionless time step.
 \mathbf{e}_z unit vector pointing in z direction.

- g gravitational acceleration (m/s^2).
- H depth of the flow domain (m).
- k isotropic permeability (m^2).
- L_c characteristic length scale (m).
- N number of pixels in images.
- p pressure (Pa).
- \bar{p} dimensionless pressure.
- P_c characteristic pressure (Pa).
- Ra Rayleigh number.
- T_c characteristic time scale (s).
- \tilde{t}_p propagation time.
- $\mathbf{u} = (u, w)$ Darcy velocity (m/s).
- $\tilde{\mathbf{u}} = (\tilde{u}, \tilde{w})$ dimensionless Darcy velocity.
- U_c characteristic velocity (m/s).
- \mathbf{w} weights of the convolutional neural network.

Greek Letters

- α weight decay.
- β_1, β_2 parameters for Adam optimization scheme.
- $\Delta\rho$ maximum fluid density contrast (kg/m^3).
- μ fluid viscosity (Pa s).
- $\rho(C)$ solute concentration dependent density (kg/m^3).
- ρ_0 initial fluid density (kg/m^3).
- ρ_{\max} maximum fluid density (kg/m^3).
- σ standard deviation of Gaussian image filter.
- ϕ porosity.

Superscripts

- est denotes estimated quantities.
- true denotes true quantities.

References

- Backhaus, S., Turitsyn, K., & Ecke, R. E. (2011). Convective instability and mass transport of diffusion layers in a Hele-Shaw geometry. *Physical Review Letters*, *106*(10), 104501. <https://doi.org/10.1103/PhysRevLett.106.104501>
- Bastian, P., Blatt, M., Dedner, A., Engwer, C., Klöforn, R., Kornhuber, R., et al. (2008b). A generic grid interface for parallel and adaptive scientific computing. Part II: Implementation and tests in DUNE. *Computing*, *82*(2), 121–138. <https://doi.org/10.1007/s00607-008-0004-9>
- Bastian, P., Blatt, M., Dedner, A., Engwer, C., Klöforn, R., Ohlberger, M., & Sander, O. (2008a). A generic grid interface for parallel and adaptive scientific computing. Part I: Abstract framework. *Computing*, *82*(2), 103–119. <https://doi.org/10.1007/s00607-008-0003-x>
- Blatt, M., & Bastian, P. (2007). The iterative solver template library. In B. Kågström, E. Elmroth, J. Dongarra, & J. Waśniewski (Eds.), *Applied parallel computing. State of the art in scientific computing* (Vol. 4699, pp. 666–675). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Blatt, M., & Bastian, P. (2008). On the generic parallelisation of iterative solvers for the finite element method. *International Journal of Computer Science & Engineering Technology*, *4*(1), 56–69. <https://doi.org/10.1504/IJCSE.2008.021112>
- Blatt, M., Burchardt, A., Dedner, A., Engwer, C., Fahlke, J., Flemisch, B., et al. (2016). The distributed and unified numerics environment, version 2.4. *Archive of Numerical Software*, *4*(100), 13–29. <https://doi.org/10.11588/ans.2016.100.26526>
- Butler, D. J., Wulff, J., Stanley, G. B., & Black, M. J. (2012). A naturalistic open source movie for optical flow evaluation. In D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, & et al. (Eds.), *Computer vision-ECCV 2012* (Vol. 7577, pp. 611–625). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-33783-3_44
- Clark, M. P., Kavetski, D., & Fenicia, F. (2011). Pursuing the method of multiple working hypotheses for hydrological modeling. *Water Resources Research*, *47*, W09301. <https://doi.org/10.1029/2010WR009827>
- de Bezenac, E., Pajot, A., & Gallinari, P. (2017). Deep learning for physical processes: Incorporating prior scientific knowledge. [arXiv:1711.07970](https://arxiv.org/abs/1711.07970) [cs, stat].
- Diersch, H.-J. G., & Kolditz, O. (2002). Variable-density flow and transport in porous media: Approaches and challenges. *Advances in Water Resources*, *25*(8–12), 899–944. [https://doi.org/10.1016/S0309-1708\(02\)00063-5](https://doi.org/10.1016/S0309-1708(02)00063-5)
- Ecke, R. E., & Backhaus, S. (2016). Plume dynamics in Hele-Shaw porous media convection. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *374*(2078), 20150420. <https://doi.org/10.1098/rsta.2015.0420>
- Ennis-King, J., & Paterson, L. (2003). Rate of dissolution due to convective mixing in the underground storage of carbon dioxide. In J. Gale & Y. Kaya (Eds.), *Greenhouse gas control technologies—6th international conference* (pp. 1653–1656). Oxford: Pergamon. <https://doi.org/10.1016/B978-008044276-1/50268-3>

Acknowledgments

The authors thank Peter Bastian at the Interdisciplinary Center for Scientific Computing (IWR), Heidelberg University, Germany, for providing the implementation of the numerical solver. The authors also thank Editor Chaopeng Shen and three anonymous reviewers for their constructive comments, which helped to improve this paper. This research is funded by the Ministerium für Wissenschaft, Forschung und Kunst Baden-Württemberg (Az 33-7533.-30-20/6/2). Hannes H. Bauser was funded in part by the Deutsche Forschungsgemeinschaft (DFG) through Project BA 6635/1-1. The underlying data are available online (<https://doi.org/10.11588/data/7NEEFK>; Kreyenberg et al., 2019).

Author Contributions

Conceptualization: P. J. K., H. H. B., and K. R.; formal analysis: P. J. K.; funding acquisition: K. R.; investigation: P. J. K.; software: P. J. K.; supervision: K. R.; validation: P. J. K.; visualization: P. J. K.; writing—original draft: P. J. K.; writing—review and editing: P. J. K., H. H. B., and K. R.

- Ennis-King, J. P., & Paterson, L. (2005). Role of convective mixing in the long-term storage of carbon dioxide in deep saline formations. *SPE Journal*, 10(03), 349–356. <https://doi.org/10.2118/84344-PA>
- Faisal, T. F., Chevalier, S., Bernabe, Y., Juanes, R., & Sassi, M. (2015). Quantitative and qualitative study of density driven CO₂ mass transfer in a vertical Hele-Shaw cell. *International Journal of Heat and Mass Transfer*, 58(11), 901–914. <https://doi.org/10.1016/j.ijheatmasstransfer.2014.11.017>
- Faisal, T. F., Chevalier, S., & Sassi, M. (2013). Experimental and numerical studies of density driven natural convection in saturated porous media with application to CO₂ geological storage. *Energy Procedia*, 37, 5323–5330. <https://doi.org/10.1016/j.egypro.2013.06.450>
- Farajzadeh, R., Salimi, H., Zitha, P. L. J., & Bruining, H. (2007). Numerical simulation of density-driven natural convection in porous media with application for CO₂ injection projects. *International Journal of Heat and Mass Transfer*, 50(25–26), 5054–5064. <https://doi.org/10.1016/j.ijheatmasstransfer.2007.08.019>
- Fernandez, J., Kurowski, P., Petitjeans, P., & Meiburg, E. (2002). Density-driven unstable flows of miscible fluids in a Hele-Shaw cell. *Journal of Fluid Mechanics*, 451, 239–260. <https://doi.org/10.1017/S0022112001006504>
- Fischer, P., Dosovitskiy, A., Ilg, E., Häusser, P., Hazirbas, C., Golkov, V., et al. (2015). FlowNet: Learning optical flow with convolutional networks. In *2015 IEEE International Conference on Computer Vision (ICCV)* (pp. 2758–2766). Santiago: IEEE. <https://doi.org/10.1109/ICCV.2015.316>
- Geiger, A., Lenz, P., Stiller, C., & Urtasun, R. (2013). Vision meets robotics: The KITTI dataset. *The International Journal of Robotics Research*, 32(11), 1231–1237. <https://doi.org/10.1177/0278364913491297>
- Hassanzadeh, H., Pooladi-Darvish, M., & Keith, D. W. (2005). Modelling of convective mixing in CO₂ storage. *Journal of Canadian Petroleum Technology*, 44(10), 43–51. <https://doi.org/10.2118/05-10-04>
- IPCC (2005). *IPCC special report on carbon dioxide capture and storage*. UK: Cambridge University Press.
- Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., & Brox, T. (2017). FlowNet 2.0: Evolution of optical flow estimation with deep networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1647–1655). Honolulu, HI: IEEE. <https://doi.org/10.1109/CVPR.2017.179>
- Janai, J., Güneş, F., Behl, A., & Geiger, A. (2017). Computer vision for autonomous vehicles: Problems, datasets and state-of-the-art. arXiv:1704.05519 [cs].
- Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., et al. (2014). Caffe: Convolutional architecture for fast feature embedding. arXiv:1408.5093 [cs].
- Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv:1412.6980 [cs].
- Kneafsey, T. J., & Pruess, K. (2010). Laboratory flow experiments for visualizing carbon dioxide-induced, density-driven brine convection. *Transport in Porous Media*, 82(1), 123–139. <https://doi.org/10.1007/s11242-009-9482-2>
- Kneafsey, T. J., & Pruess, K. (2011). Laboratory experiments and numerical simulation studies of convectively enhanced carbon dioxide dissolution. *Energy Procedia*, 4, 5114–5121. <https://doi.org/10.1016/j.egypro.2011.02.487>
- Kolditz, O., Ratke, R., Diersch, H. G., & Zielke, W. (1998). Coupled groundwater flow and transport: 1. Verification of variable density flow and transport models. *Advances in Water Resources*, 21(1), 27–46. [https://doi.org/10.1016/S0309-1708\(96\)00034-6](https://doi.org/10.1016/S0309-1708(96)00034-6)
- Kreyenberg, P. J., Bauser, H. H., & Roth, K. (2019). Velocity field estimation on density-driven solute transport with a convolutional neural network [dataset] heiDATA. <https://doi.org/10.11588/data/7NEEF>
- Lindeberg, E., & Wessel-Berg, D. (1997). Vertical convection in an aquifer column under a gas cap of CO₂. *Energy Conversion and Management*, 38, S229–S234. [https://doi.org/10.1016/S0196-8904\(96\)00274-9](https://doi.org/10.1016/S0196-8904(96)00274-9)
- Liu, Q. (2017). Stein variational gradient descent as gradient flow. *Advances in neural information processing systems*, 3115–3123.
- Liu, Q., & Wang, D. (2016). Stein variational gradient descent: A general purpose Bayesian inference algorithm. *Advances in neural information processing systems*, 2378–2386.
- Maas, A. L., Hannun, A. Y., & Ng, A. Y. (2013). Rectifier nonlinearities improve neural network acoustic models. *International Conference on Machine Learning*, 30, 3.
- Marçais, J., & de Dreuzy, J.-R. (2017). Prospective interest of deep learning for hydrological inference. *Groundwater*, 55(5), 688–692. <https://doi.org/10.1111/gwat.12557>
- Mayer, N., Ilg, E., Häusser, P., Fischer, P., Cremers, D., Dosovitskiy, A., & Brox, T. (2016). A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 4040–4048) Las Vegas, NV. <https://doi.org/10.1109/CVPR.2016.438>
- Mo, S., Zhu, Y., Zabarar, N., Shi, X., & Wu, J. (2019). Deep convolutional encoder decoder networks for uncertainty quantification of dynamic multiphase flow in heterogeneous media. *Water Resources Research*, 55, 703–728. <https://doi.org/10.1029/2018WR023528>
- Nowak, W., & Cirpka, O. A. (2006). Geostatistical inference of hydraulic conductivity and dispersivities from hydraulic heads and tracer data. *Water Resources Research*, 42, W08416. <https://doi.org/10.1029/2005WR004832>
- Oltean, C., Felder, Ch., Panfilov, M., & Buès, M. A. (2004). Transport with a very low density contrast in Hele-Shaw cell and porous medium: Evolution of the mixing zone. *Transport in Porous Media*, 55(3), 339–360. <https://doi.org/10.1023/B:TIPM.0000013332.08029.af>
- Pau, G. S., Bell, J. B., Pruess, K., Almgren, A. S., Lijewski, M. J., & Zhang, K. (2010). High-resolution simulation and characterization of density-driven flow in CO₂ storage in saline aquifers. *Advances in Water Resources*, 33(4), 443–455. <https://doi.org/10.1016/j.advwatres.2010.01.009>
- Pruess, K., & Zhang, K. (2008). Numerical modeling studies of the dissolution-diffusion-convection process during CO₂ storage in saline aquifers (LBNL-1243E, 944124). CA, USA. <https://doi.org/10.2172/944124>
- Rasmusson, M., Fagerlund, F., Rasmusson, K., Tsang, Y., & Niemi, A. (2017). Refractive-light-transmission technique applied to density-driven convective mixing in porous media with implications for geological CO₂ storage. *Water Resources Research*, 53, 8760–8780. <https://doi.org/10.1002/2017WR020730>
- Riaz, A., Hesse, M., Tchelepi, H. A., & Orr, F. M. (2006). Onset of convection in a gravitationally unstable diffusive boundary layer in porous media. *Journal of Fluid Mechanics*, 548, 87–111. <https://doi.org/10.1017/S0022112005007494>
- Shen, C. (2018). A transdisciplinary review of deep learning research and its relevance for water resources scientists. *Water Resources Research*, 54, 8558–8593. <https://doi.org/10.1029/2018WR022643>
- Shen, C., Laloy, E., Elshorbagy, A., Albert, A., Bales, J., Chang, F.-J., et al. (2018). Hess opinions: Incubating deep-learning-powered hydrologic science advances as a community. *Hydrology and Earth System Sciences*, 22(11), 5639–5656. <https://doi.org/10.5194/hess-22-5639-2018>
- Simonyan, K., & Zisserman, A. (2014a). Two-stream convolutional networks for action recognition in videos. arXiv:1406.2199 [cs].
- Simonyan, K., & Zisserman, A. (2014b). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 [cs].

- Slim, A. C., Bandi, M. M., Miller, J. C., & Mahadevan, L. (2013). Dissolution-driven convection in a Hele-Shaw cell. *Physics of Fluids*, *25*(2), 024101. <https://doi.org/10.1063/1.4790511>
- Thomas, C., Dehaeck, S., & De Wit, A. (2018). Convective dissolution of CO₂ in water and salt solutions. *International Journal of Greenhouse Gas Control*, *72*, 105–116. <https://doi.org/10.1016/j.ijggc.2018.01.019>
- Thomas, C., Lemaigre, L., Zalts, A., D'Onofrio, A., & De Wit, A. (2015). Experimental study of CO₂ convective dissolution: The effect of color indicators. *International Journal of Greenhouse Gas Control*, *42*, 525–533. <https://doi.org/10.1016/j.ijggc.2015.09.002>
- Vogel, H.-J., Bartke, S., Daedlow, K., Helming, K., Kögel-Knabner, I., Lang, B., et al. (2018). A systemic approach for modeling soil functions. *Soil*, *4*(1), 83–92. <https://doi.org/10.5194/soil-4-83-2018>
- Weir, G. J., White, S. P., & Kissling, W. M. (1995). Reservoir storage and containment of greenhouse gases. *Energy Conversion and Management*, *36*(6), 531–534. [https://doi.org/10.1016/0196-8904\(95\)00060-Q](https://doi.org/10.1016/0196-8904(95)00060-Q)
- Wooding, R. A., Tyler, S. W., & White, I. (1997). Convection in groundwater below an evaporating salt lake: 1. Onset of instability. *Water Resources Research*, *33*(6), 1199–1217. <https://doi.org/10.1029/96WR03533>
- Wooding, R. A., Tyler, S. W., White, I., & Anderson, P. A. (1997). Convection in groundwater below an evaporating salt lake: 2. Evolution of fingers or plumes. *Water Resources Research*, *33*(6), 1219–1228. <https://doi.org/10.1029/96WR03534>
- Yang, C., & Gu, Y. (2006). Accelerated mass transfer of CO₂ in reservoir brine due to density-driven natural convection at high pressures and elevated temperatures. *Industrial & Engineering Chemistry Research*, *45*(8), 2430–2436. <https://doi.org/10.1021/ie050497r>
- Zhu, Y., & Zabaras, N. (2018). Bayesian deep convolutional encoder-decoder networks for surrogate modeling and uncertainty quantification. *Journal of Computational Physics*, *366*, 415–447. <https://doi.org/10.1016/j.jcp.2018.04.018>