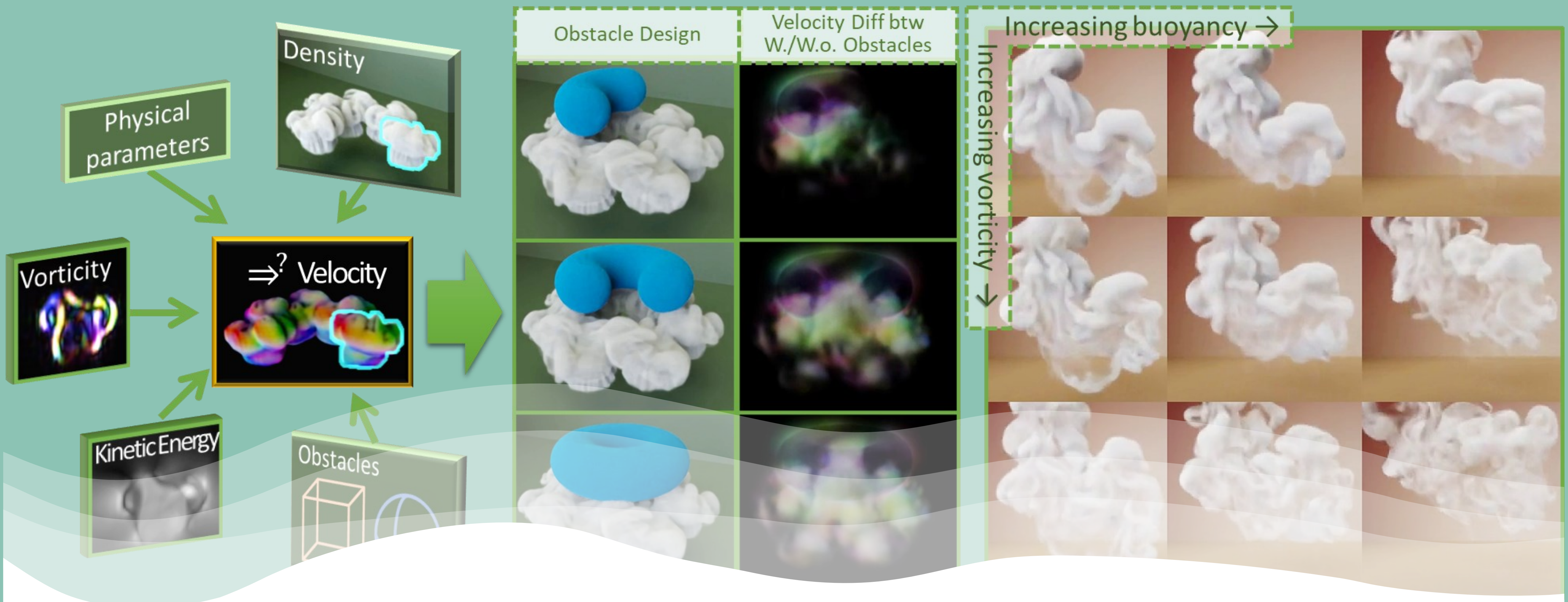


Learning Meaningful Controls for Fluids

Mengyu Chu[1], Nils Thuerey[2], Hans-Peter Seidel[1], Christian Theobalt[1] , and Rhaleb Zayer[1]

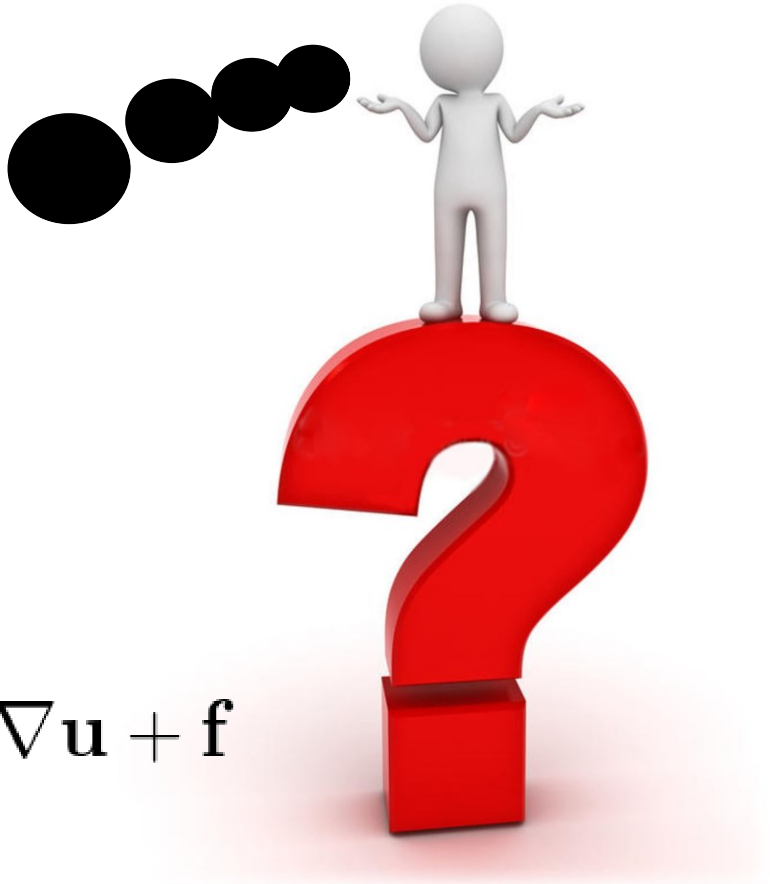
1 Max Planck Institute for Informatics, Saarland Informatics Campus 2 Technical University of Munich

max planck institut informatik

Technical University of Munich

Motivation

Fluid simulations

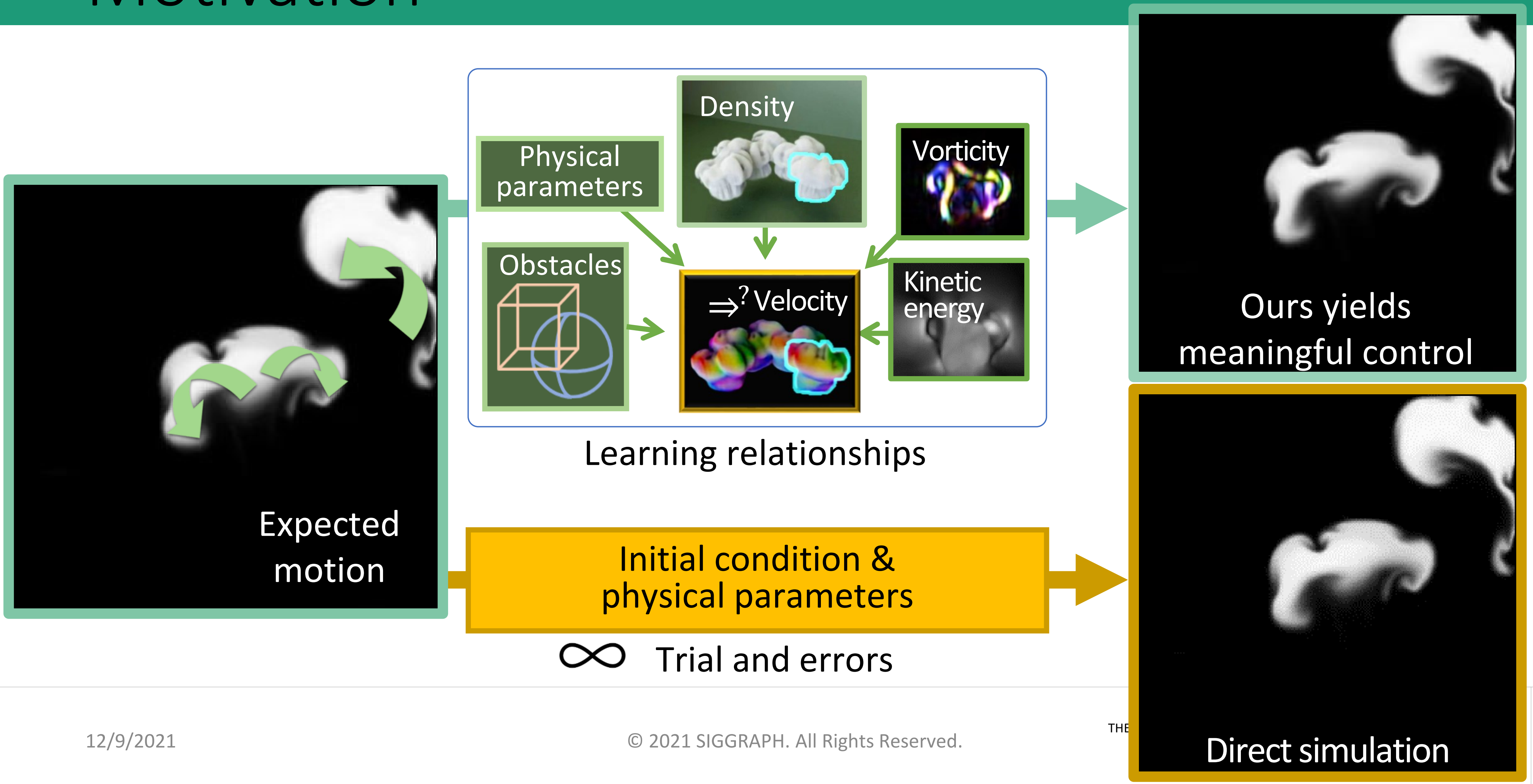


Navier-Stokes equations

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{1}{\rho} \nabla p + \nu \nabla \cdot \nabla \mathbf{u} + \mathbf{f}$$

$$\nabla \cdot \mathbf{u} = 0$$

Motivation


Expected motion
Physical parameters
Density
Vorticity
Obstacles
$\Rightarrow^{?}$ Velocity
Kinetic energy
Learning relationships
Ours yields meaningful control
Initial condition & physical parameters
∞ Trial and errors
Direct simulation

Related Work

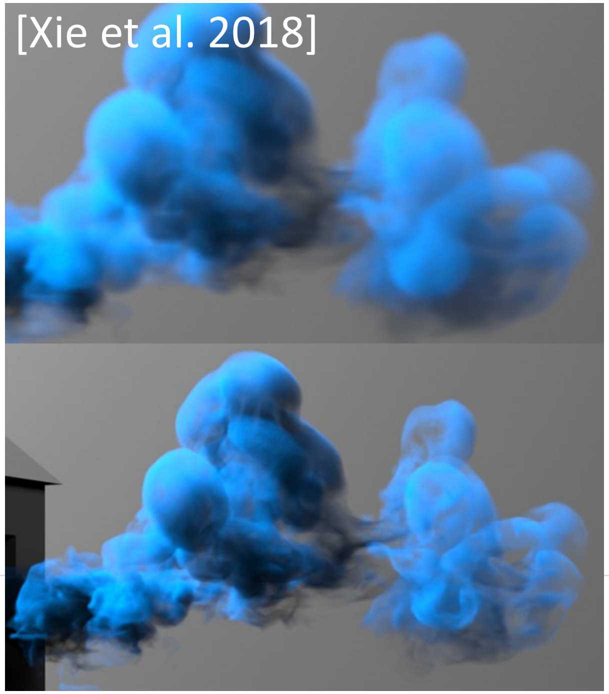
- Flexible fluid manipulations for users:
  - Fluid control to match target distributions
  - Fluid guiding to match coarse target
  - Detail synthesis to add fine features

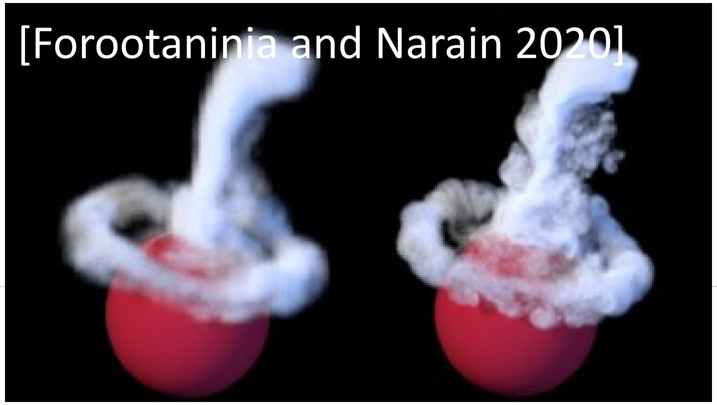
Ours: a simulation method with visual manipulations

- Deep learning algorithms
  - GANs [Goodfellow et al. 2014]
  - Conditional GANs
  - Fighting against mode collapse

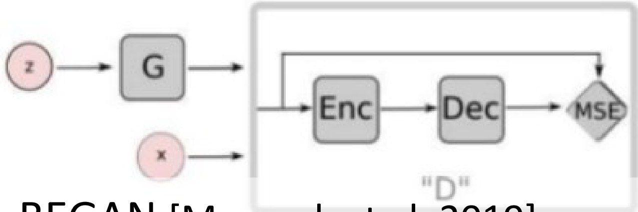
Ours: improved control sensitivity


[Pan and Manocha 2017]

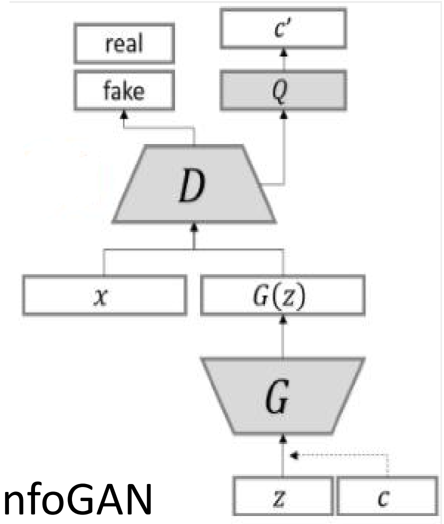

[Xie et al. 2018]


[Forootaninia and Narain 2020]


Pix2Pix [Isola et al. 2017]

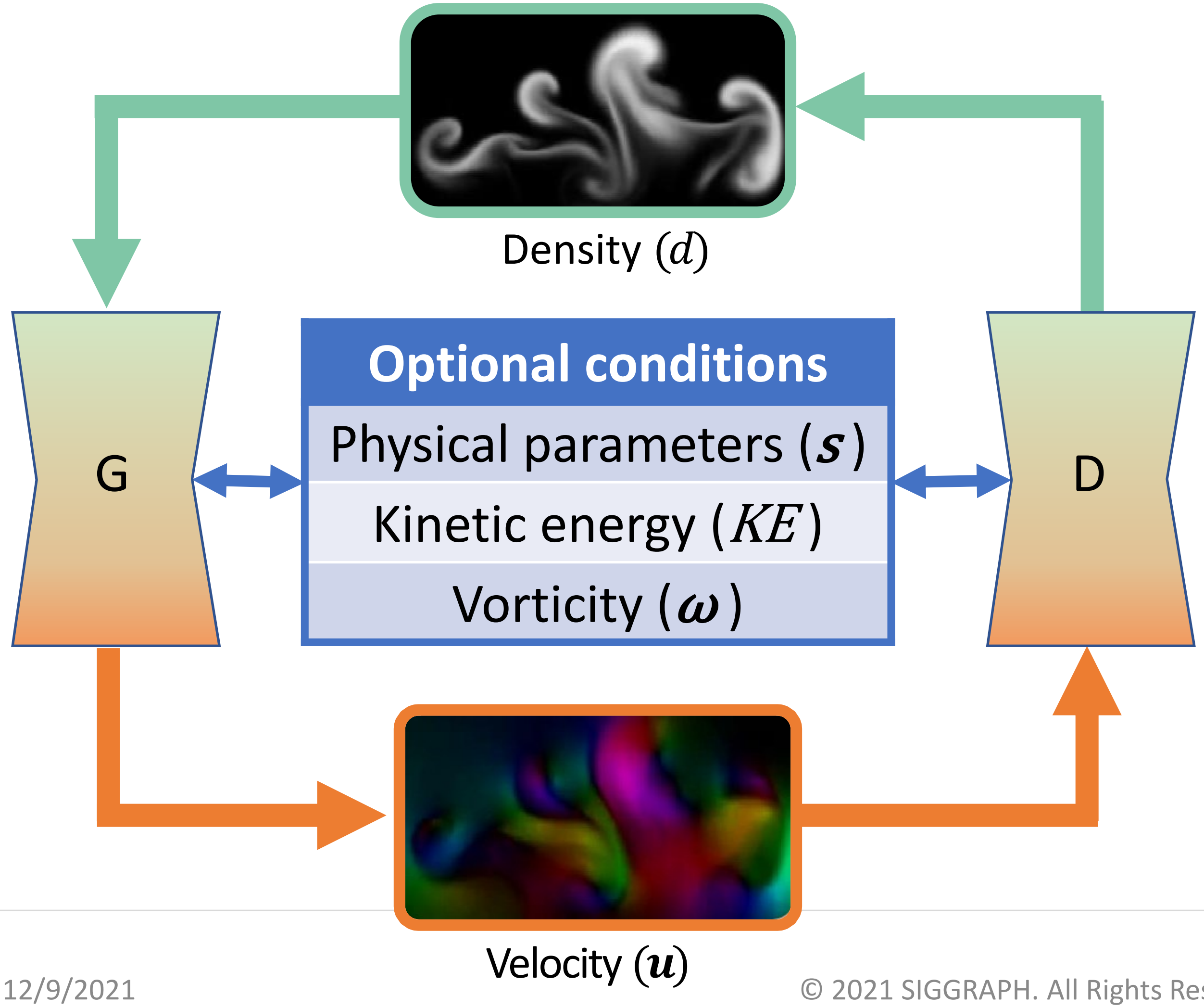


BEGAN [Marzouk et al. 2019]



InfoGAN [Chen et al. 2016]

Method

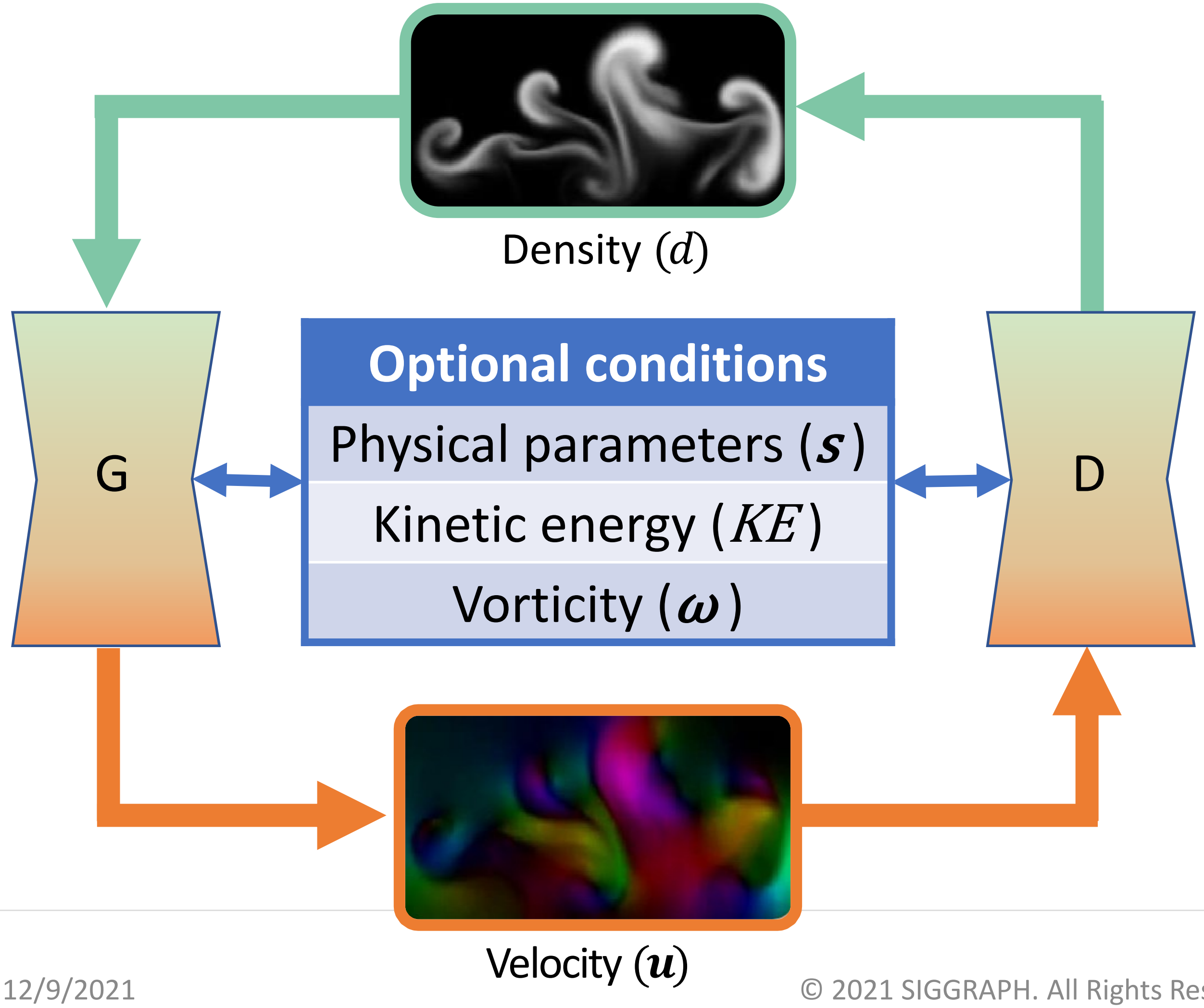


Our simulation:

$$\boldsymbol{u}_t = \text{Generator}\,(d_t, \dots)$$

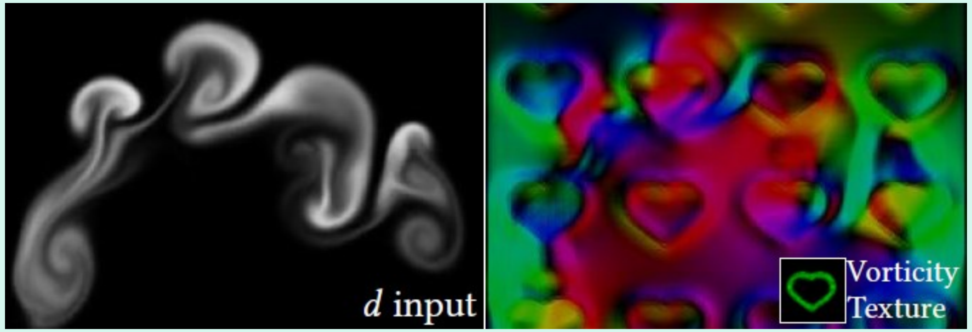
$$d_{t+1} = \text{Advection}\,(d_t, \boldsymbol{u}_t)$$

Method

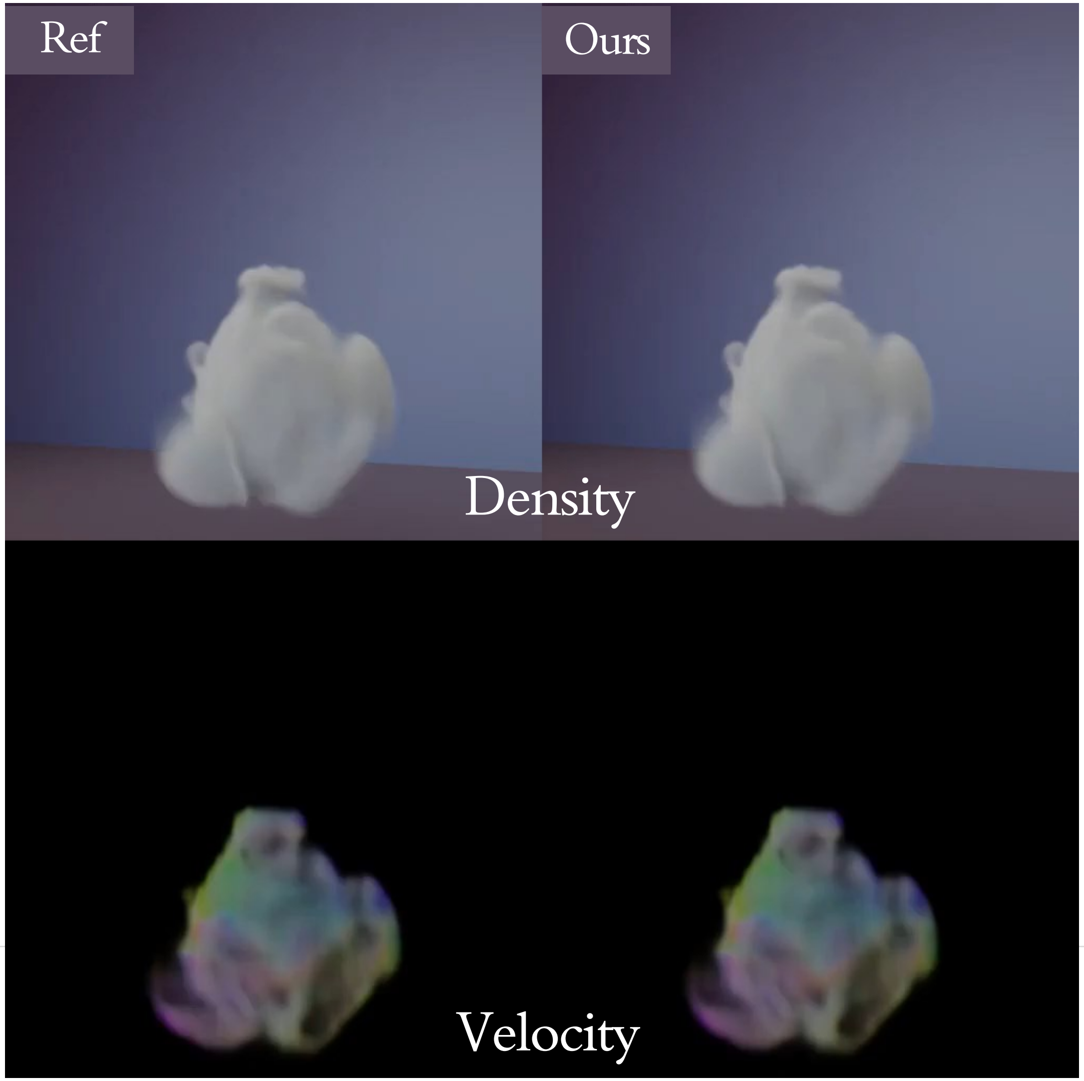


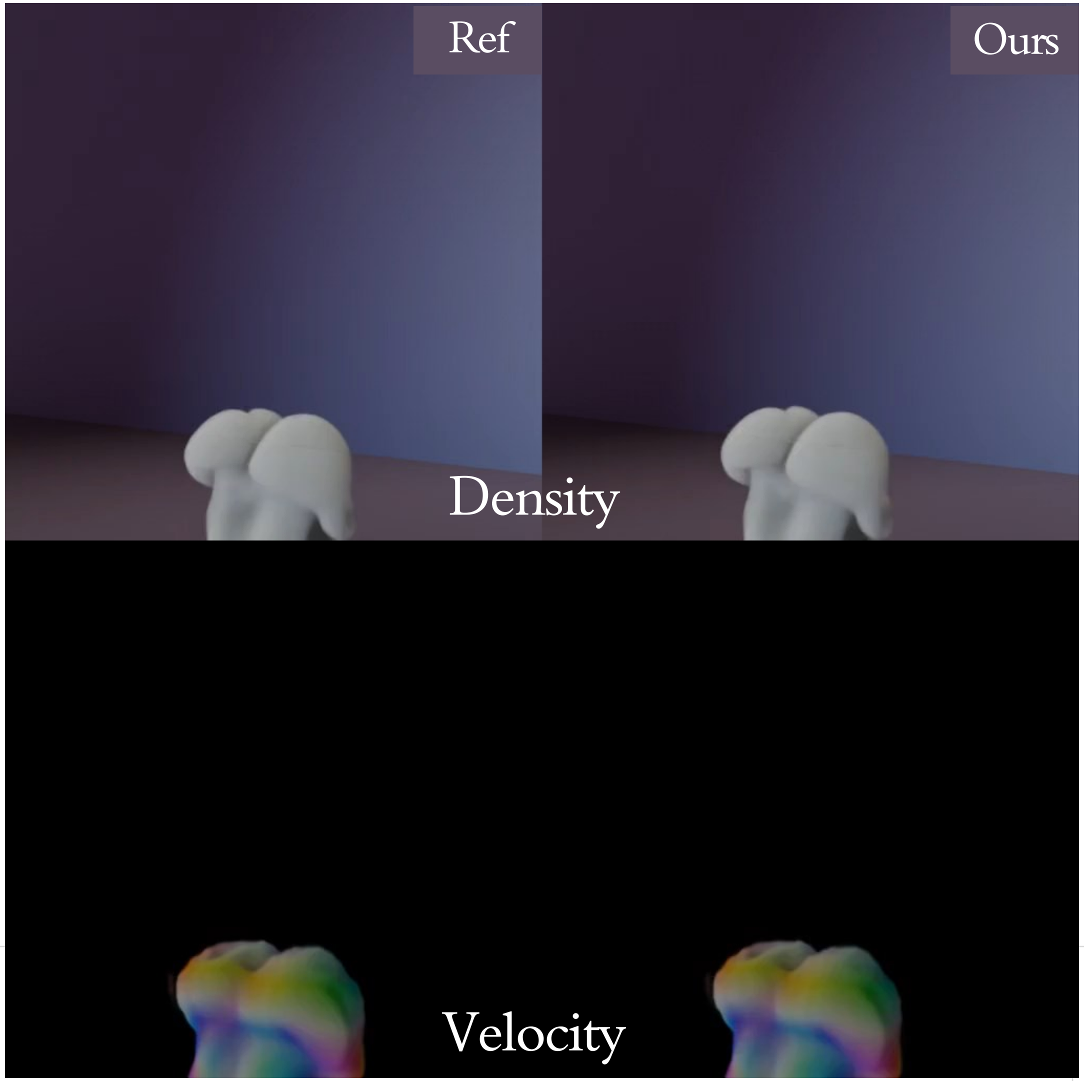
Latent-space controls:

$$\boldsymbol{u}_t = \mathrm{G}\,(d_t, \boldsymbol{s}, \mathrm{KE}, \boldsymbol{\omega})$$

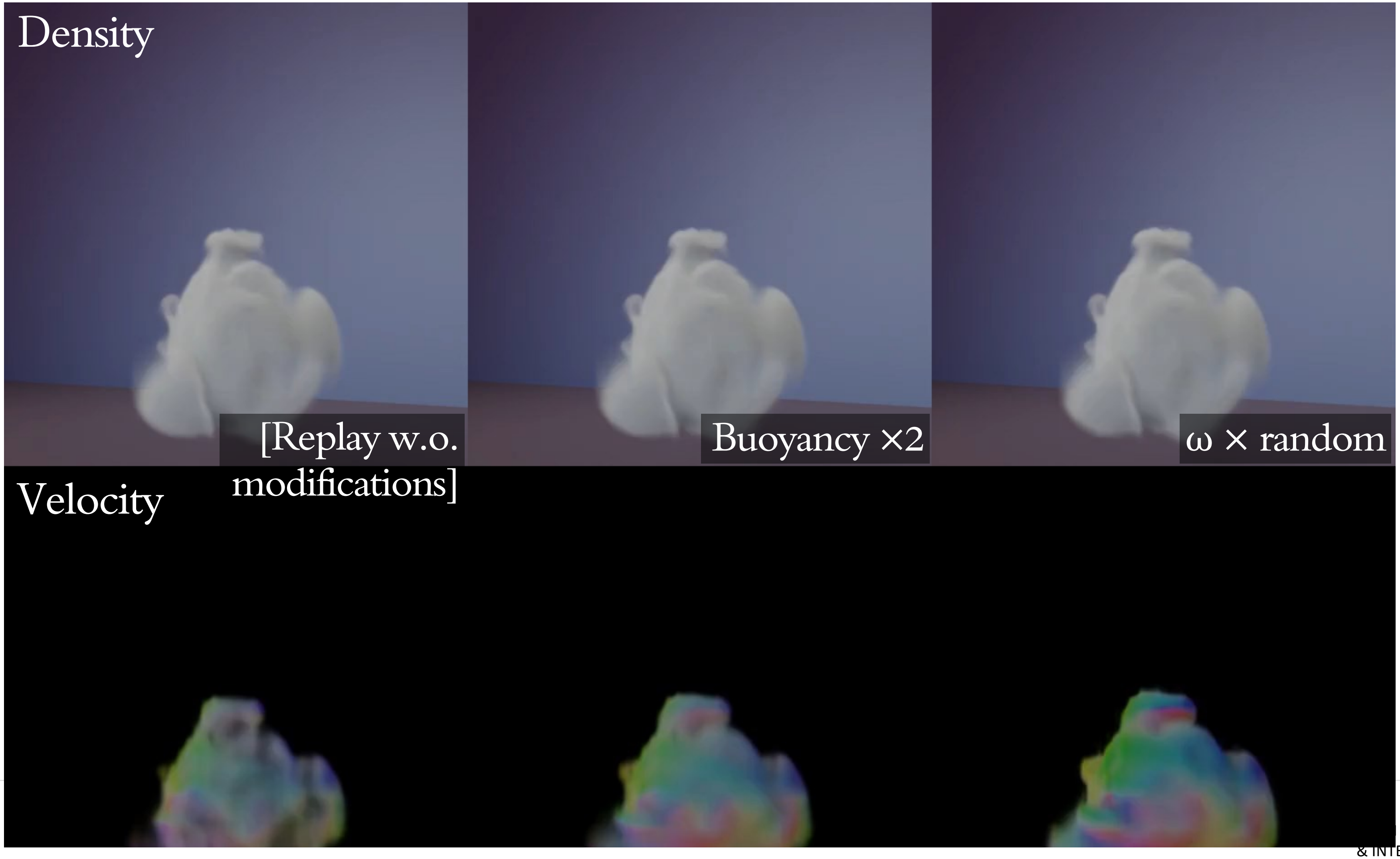


3D Results Based on Initial Density

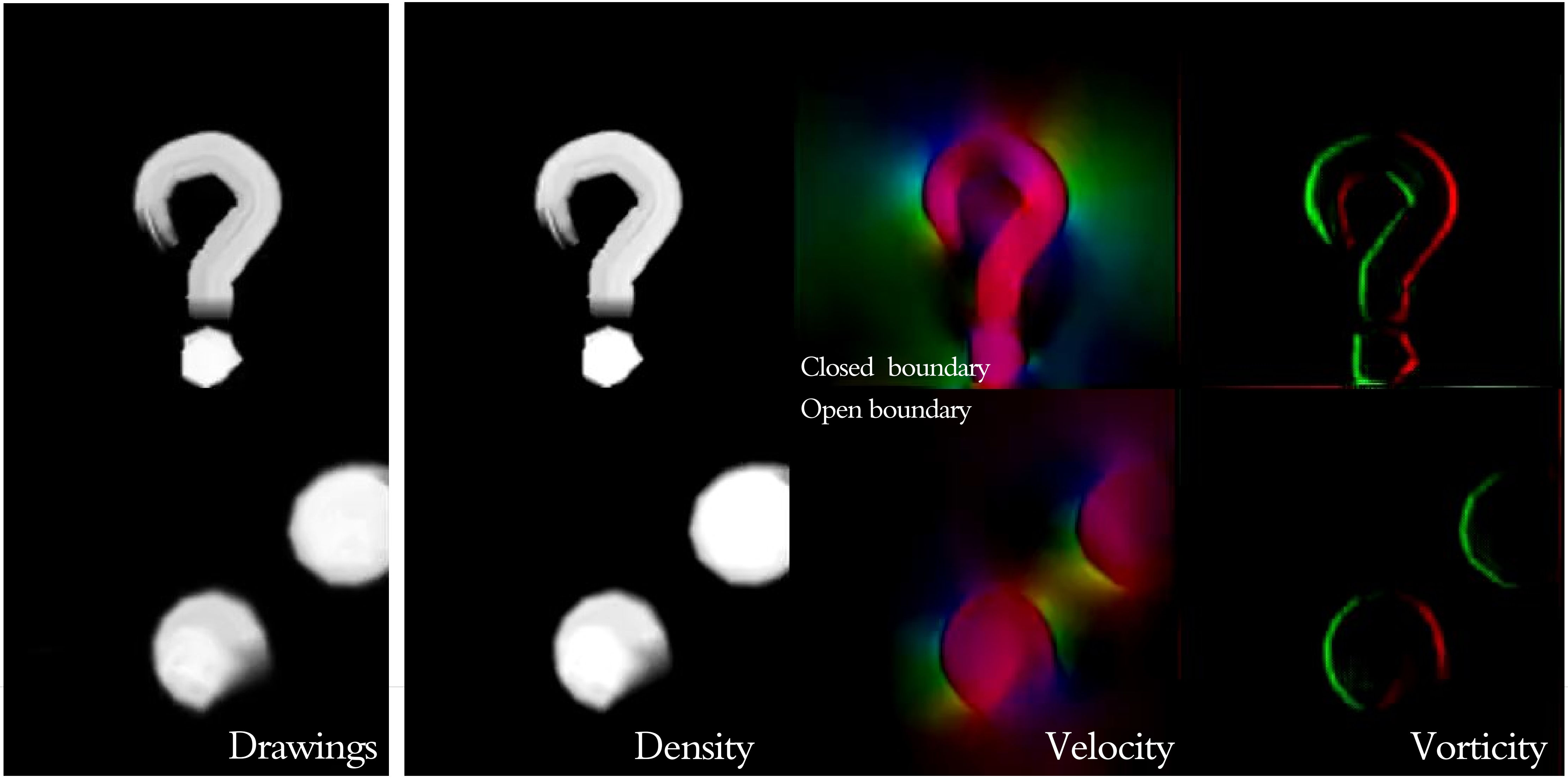




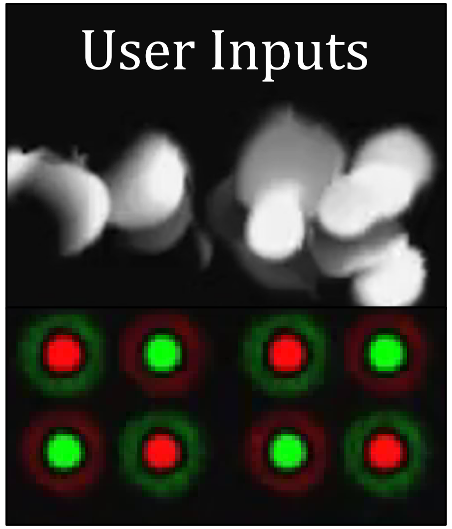
3D Results with Modifications

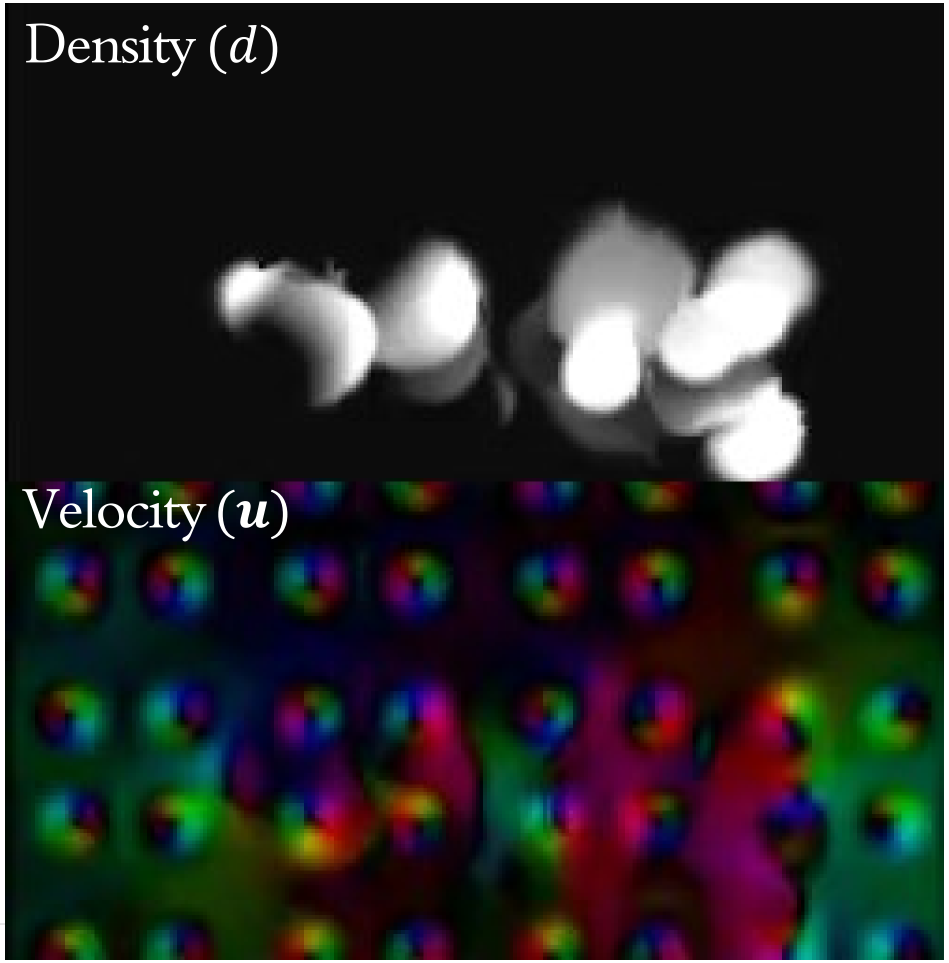


Generalization to Drawings

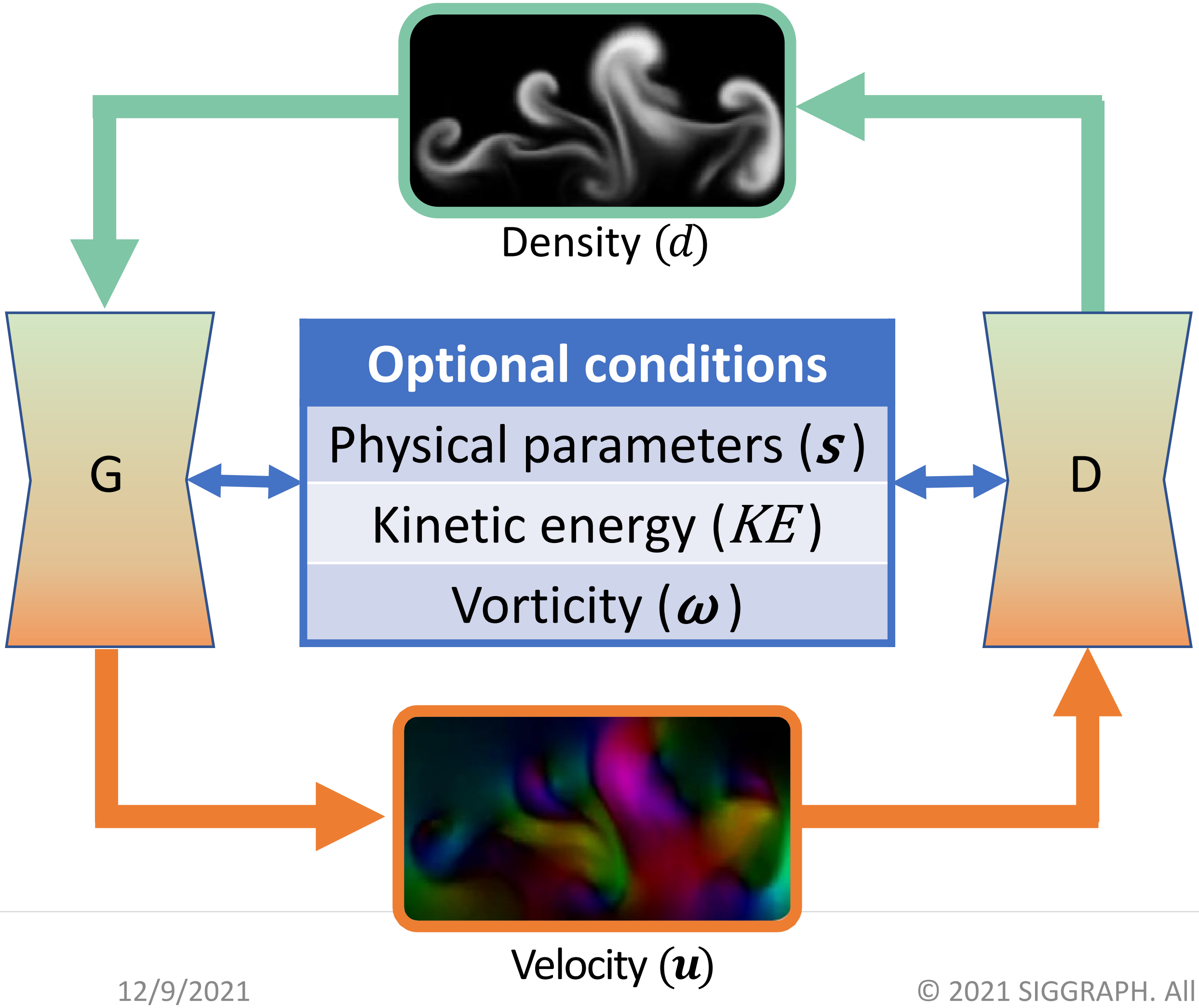


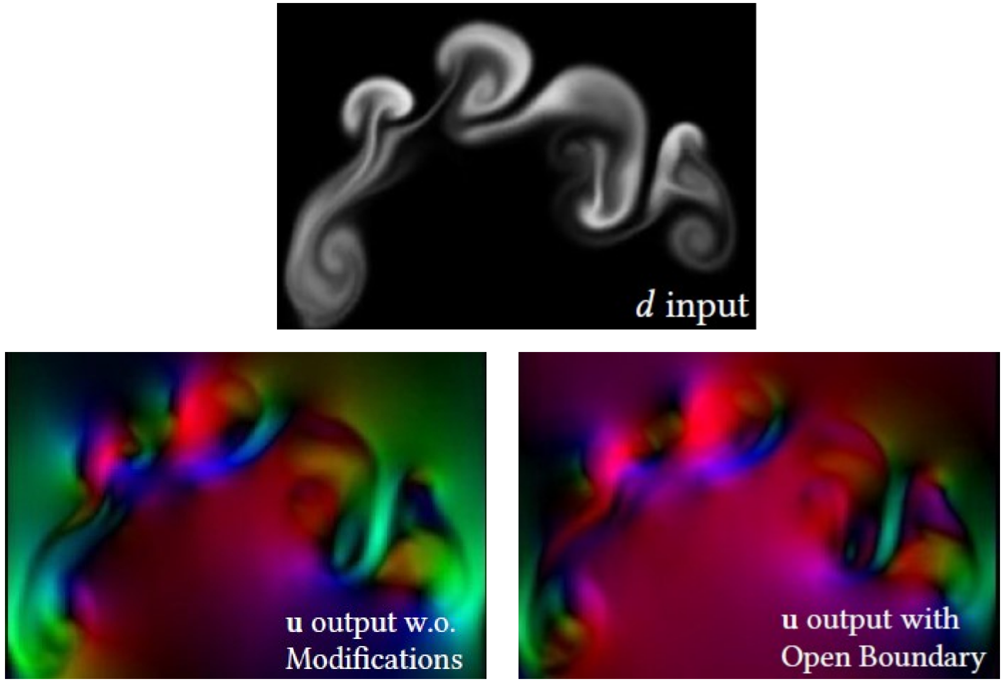
Generalization to Texture-Based Controls



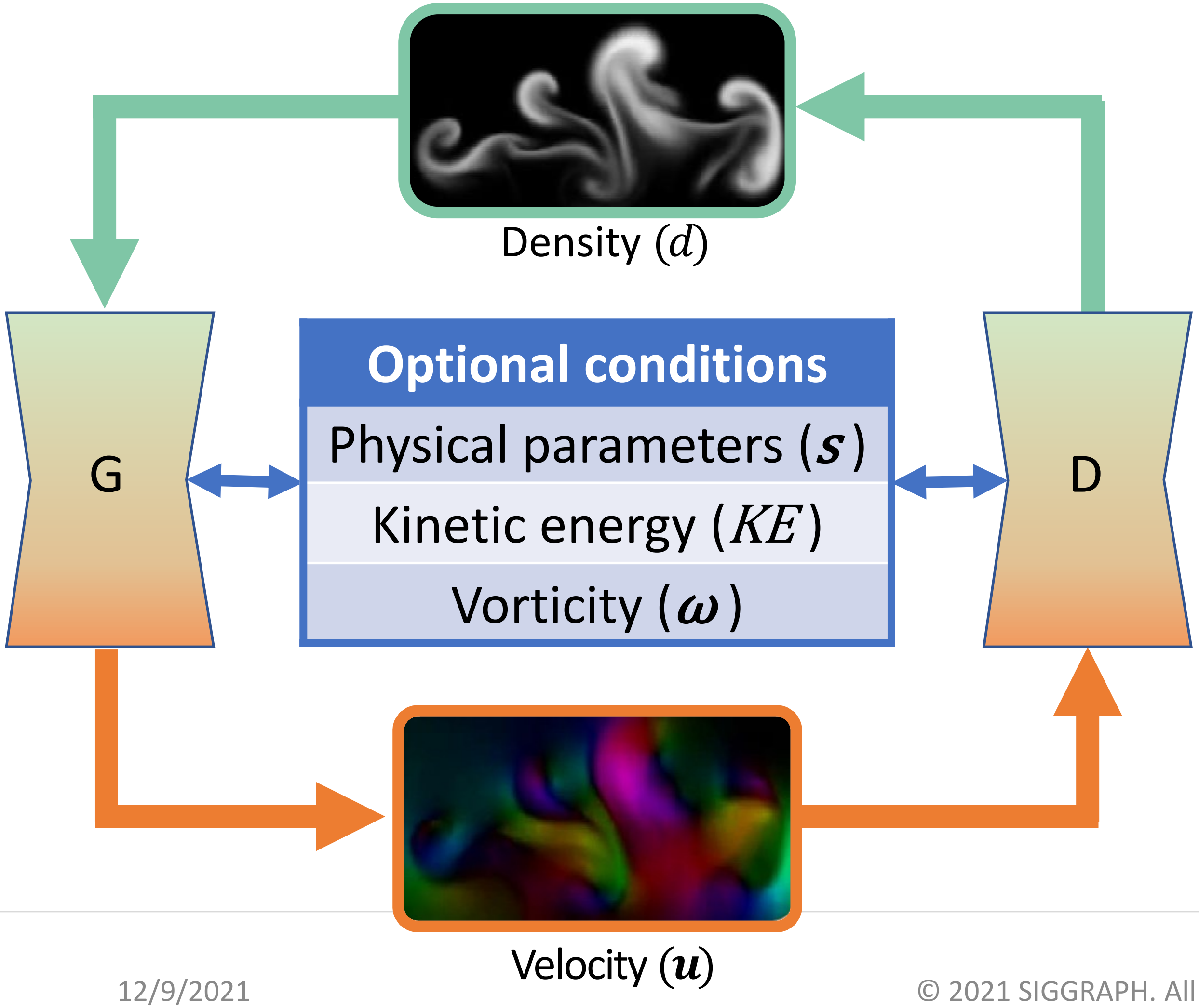


Method


Density (d)
G
Optional conditions
Physical parameters (s)
Kinetic energy (KE)
Vorticity (ω)
D
Velocity (u)


d input
u output w.o. Modifications
u output with Open Boundary

Method



Control Disentanglement:

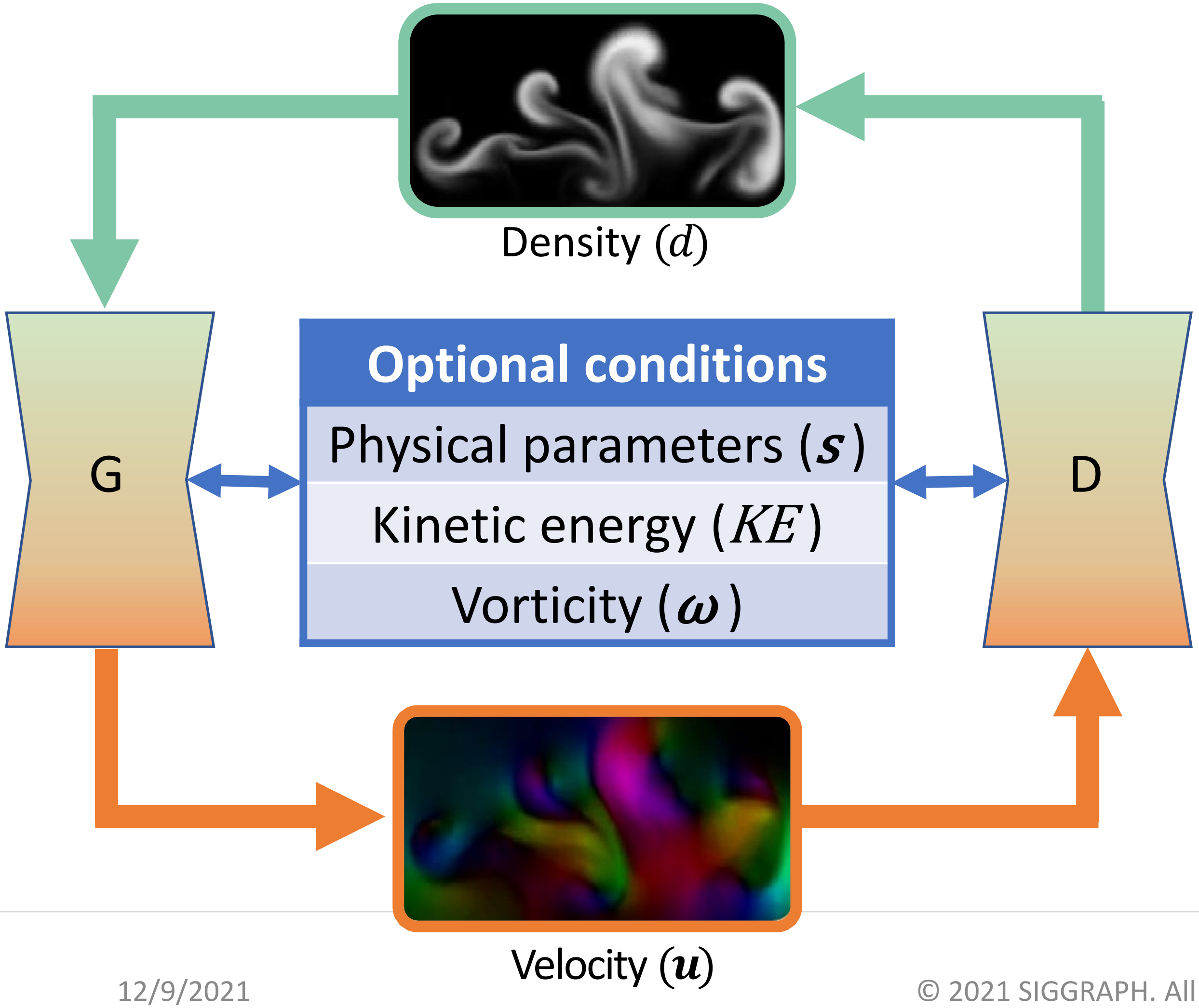
$$\begin{cases} \boldsymbol{u}_t = \mathrm{G}\,(d_t, \underline{\boldsymbol{s}, \mathrm{KE}, \boldsymbol{\omega}}\,) \\ d_t, \boldsymbol{s}, \mathrm{KE}, \boldsymbol{\omega} = \mathrm{D}\,(\boldsymbol{u}_t) \end{cases}$$

$$\mathcal{L}_{G,\mathrm{Restore}} = \|\boldsymbol{u}_t - \mathrm{G}(d_t)\| + \|d_t - \mathrm{D}(\mathrm{G}(d_t))\|$$

$$\mathcal{L}_{\mathrm{D}} = \|d_t - \mathrm{D}(\boldsymbol{u}_t)\| - \|d_t - \mathrm{D}(\mathrm{G}(d_t))\|$$

$$\mathcal{L}_{G,\mathrm{Mod}} = \|(d_t, \mathrm{s}', \mathrm{KE}', \boldsymbol{\omega}') - \mathrm{D}(\mathrm{G}(d_t, \mathrm{s}', \mathrm{KE}', \boldsymbol{\omega}'))\|$$

Method



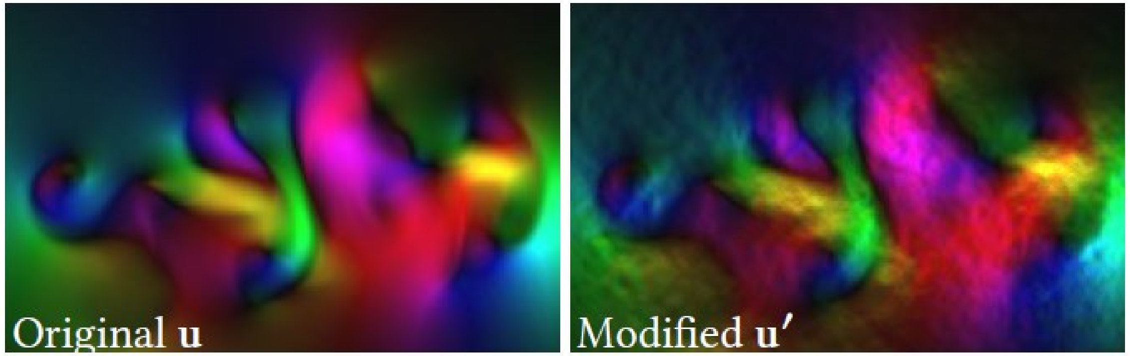
$$\mathcal{L}_G = \mathcal{L}_{G,\text{Restore}} + \mathcal{L}_{G,\text{Mod}}$$

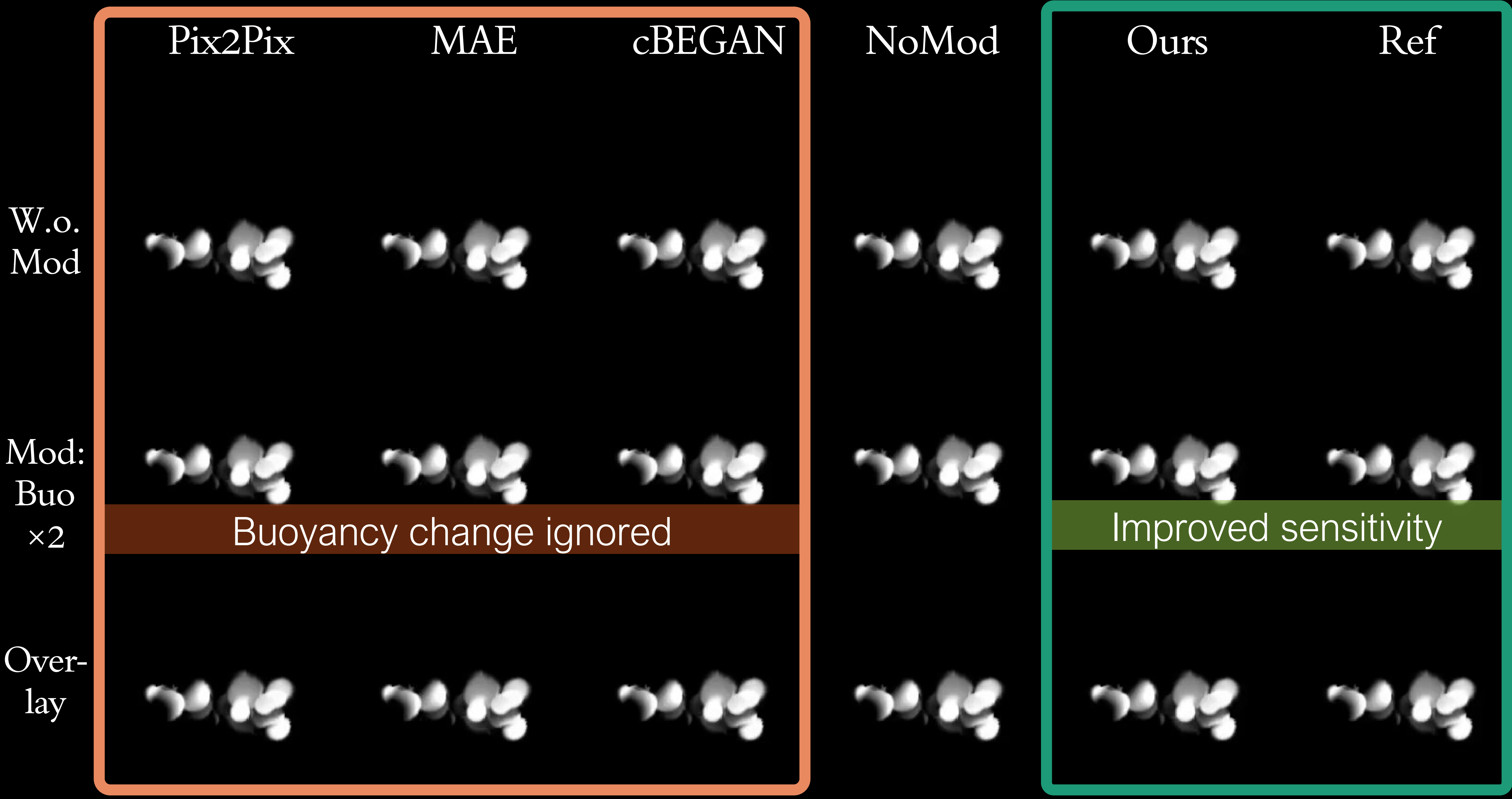
$$\mathcal{L}_{G,\text{Mod}} = \|(d_t, \boldsymbol{s}', \text{KE}', \boldsymbol{\omega}') - \text{D}(\text{G}(d_t, \underline{\boldsymbol{s}', \text{KE}', \boldsymbol{\omega}'}))\|$$

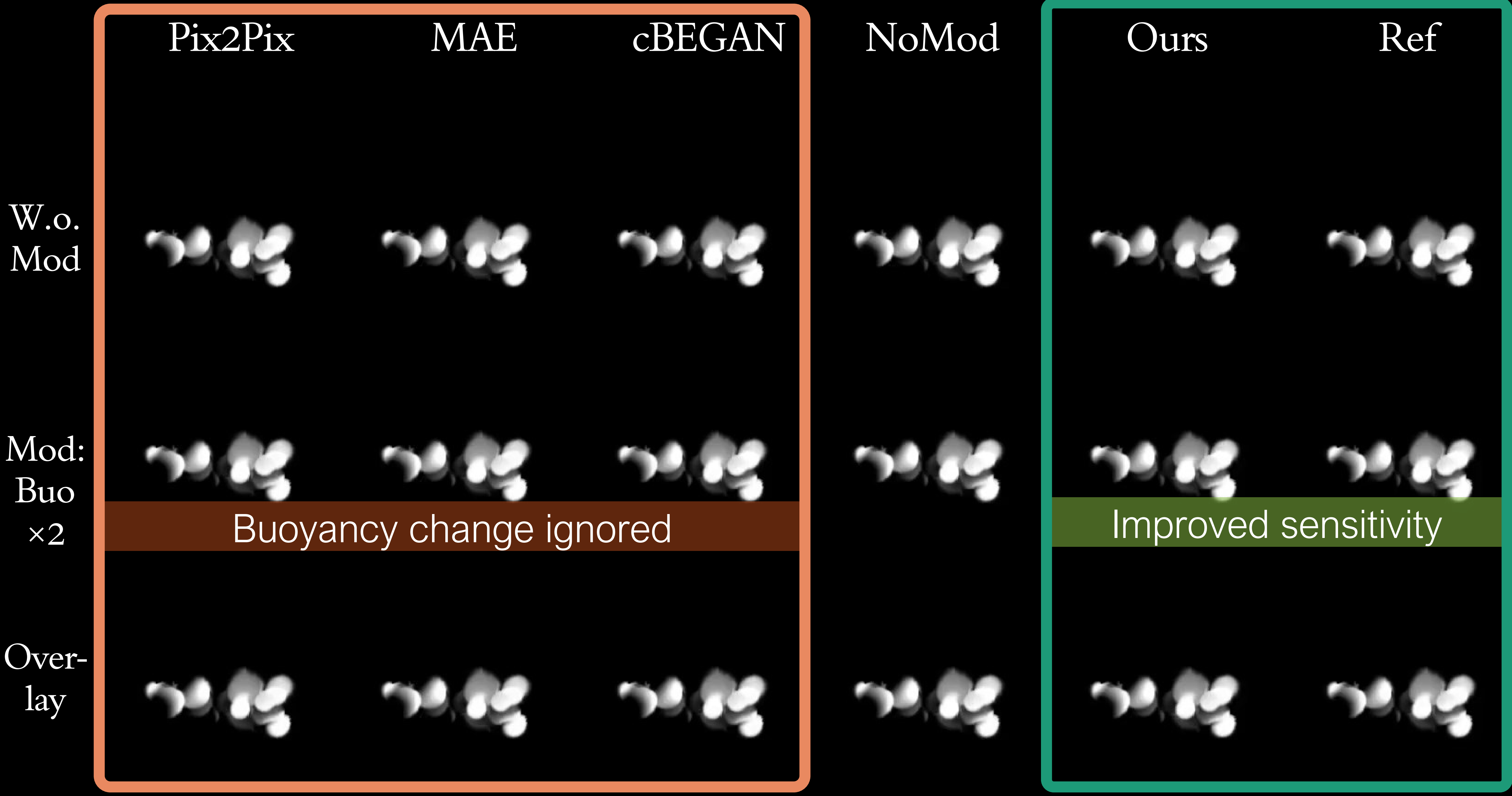
$\boldsymbol{s}'$: sampled from the training range

$\text{KE}', \boldsymbol{\omega}'$: calculated from a modified velocity

with wavelet turbulence [Kim et al. 2008]




Pix2Pix
MAE
cBEGAN
NoMod
Ours
Ref
W.o. Mod
Mod: Buo ×2
Over-lay
Buoyancy change ignored
Improved sensitivity



Pix2Pix
MAE
cBEGAN
NoMod
Ours
Ref
W.o. Mod
Mod: Buo ×2
Over-lay
Buoyancy change ignored
Improved sensitivity

Ours, buoyancy ×2

References, varying buoyancy: ×.5 ×1. ×1.5 ×2 ×2.5

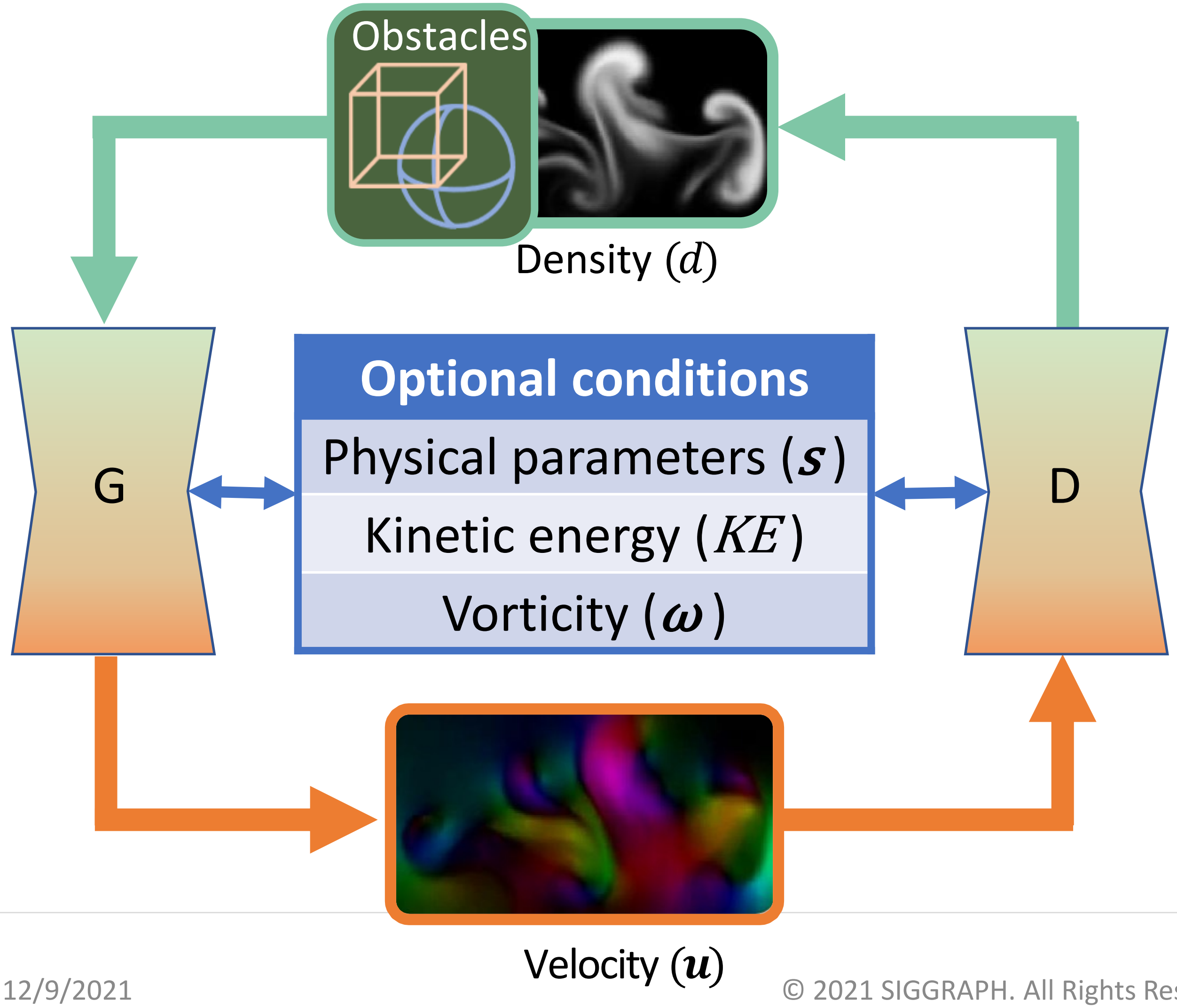
d

u

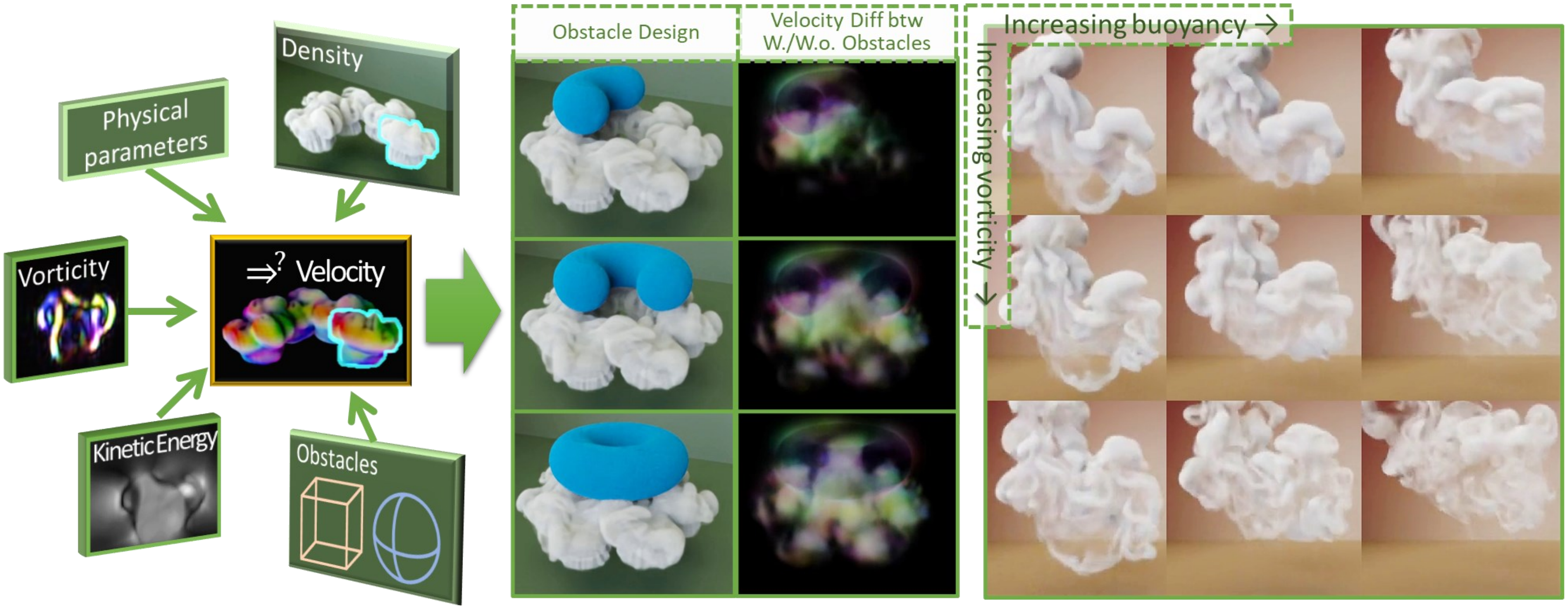
Avg. similarity on 150 frames	Our-2.0, Ref-0.5	Our-2.0, Ref-1.0	Our-2.0, Ref-1.5	**Our-2.0, Ref-2.0**	Our-2.0, Ref-2.5
	$\not\approx$	$\not\approx$	$\not\approx$	$\approx$	$\not\approx$
LSiM↓ [Kohl et al.]	0.295	0.225	0.168	0.137	0.160

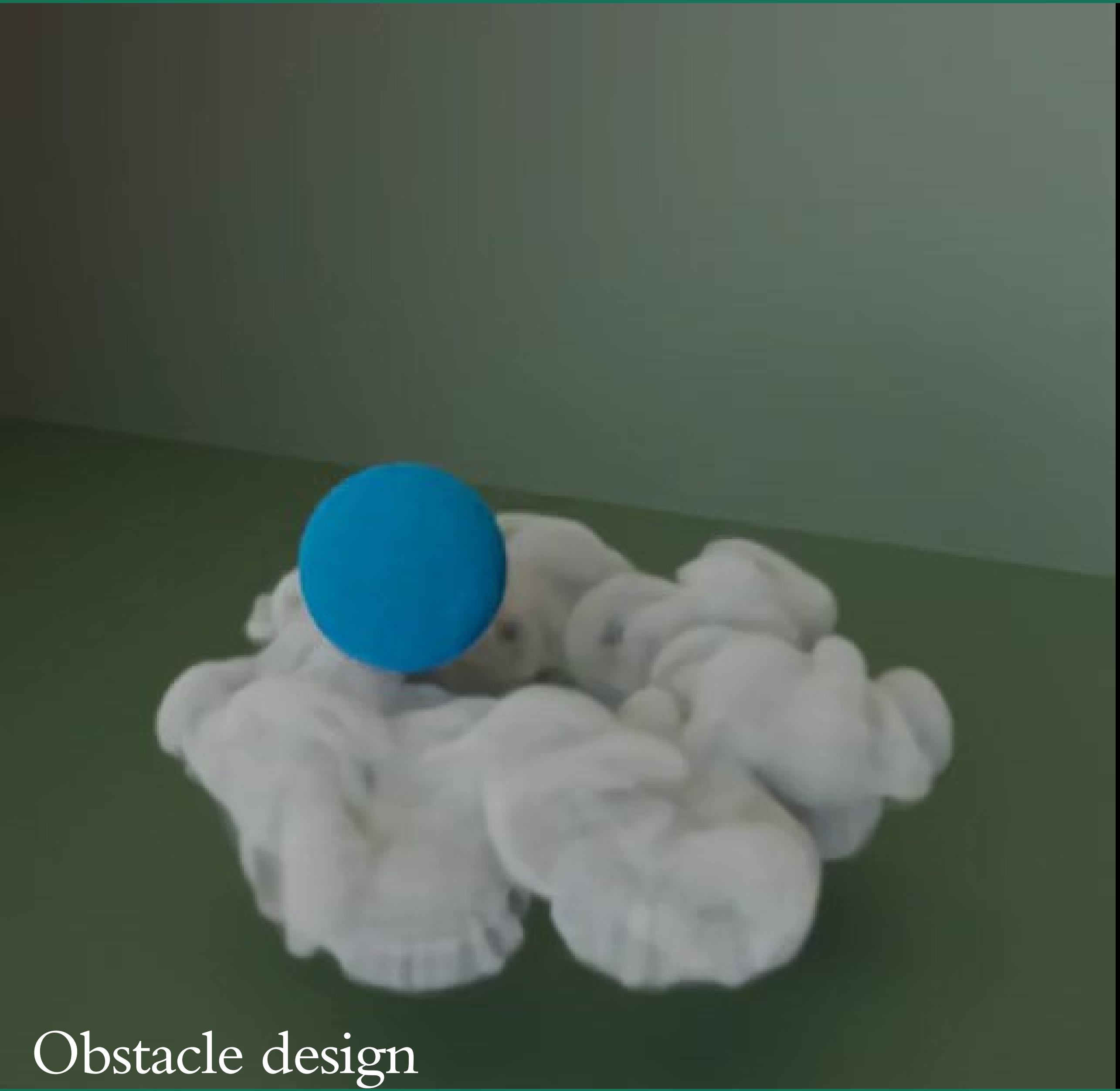
ω

Method

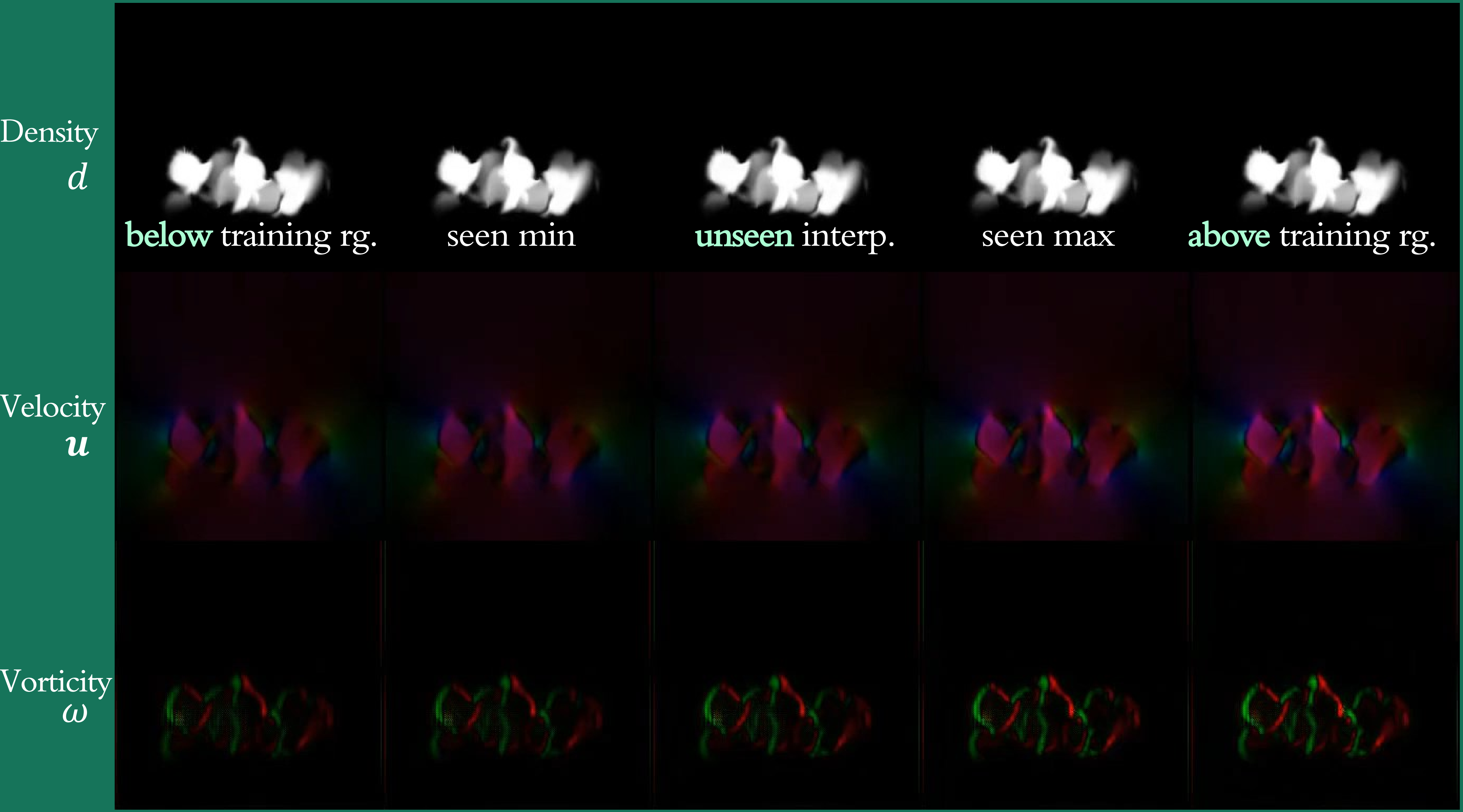

Obstacles
Density (d)
G
D
Optional conditions
Physical parameters (s)
Kinetic energy (KE)
Vorticity (ω)
Velocity $(\boldsymbol{u})$

Results and Use Cases

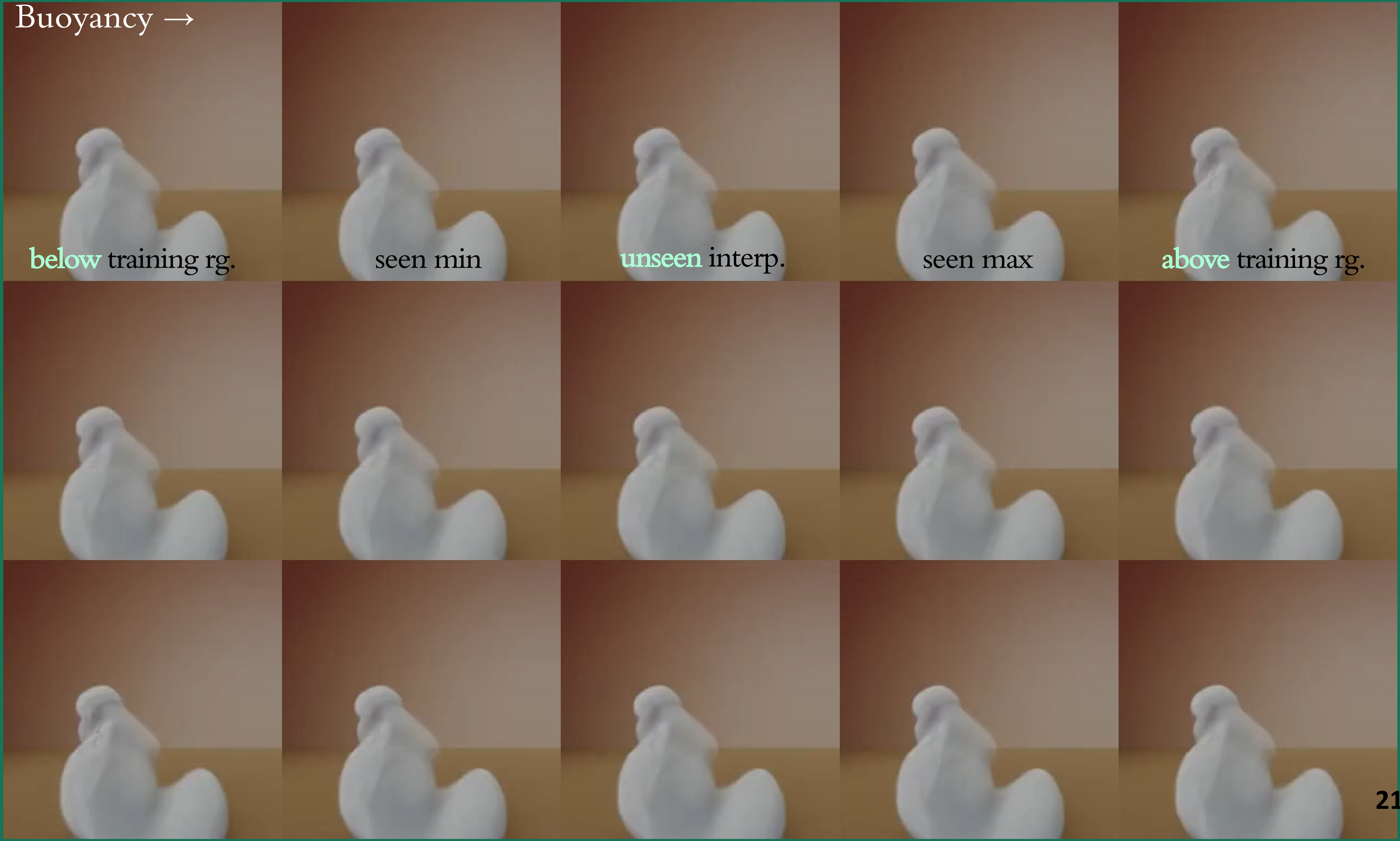


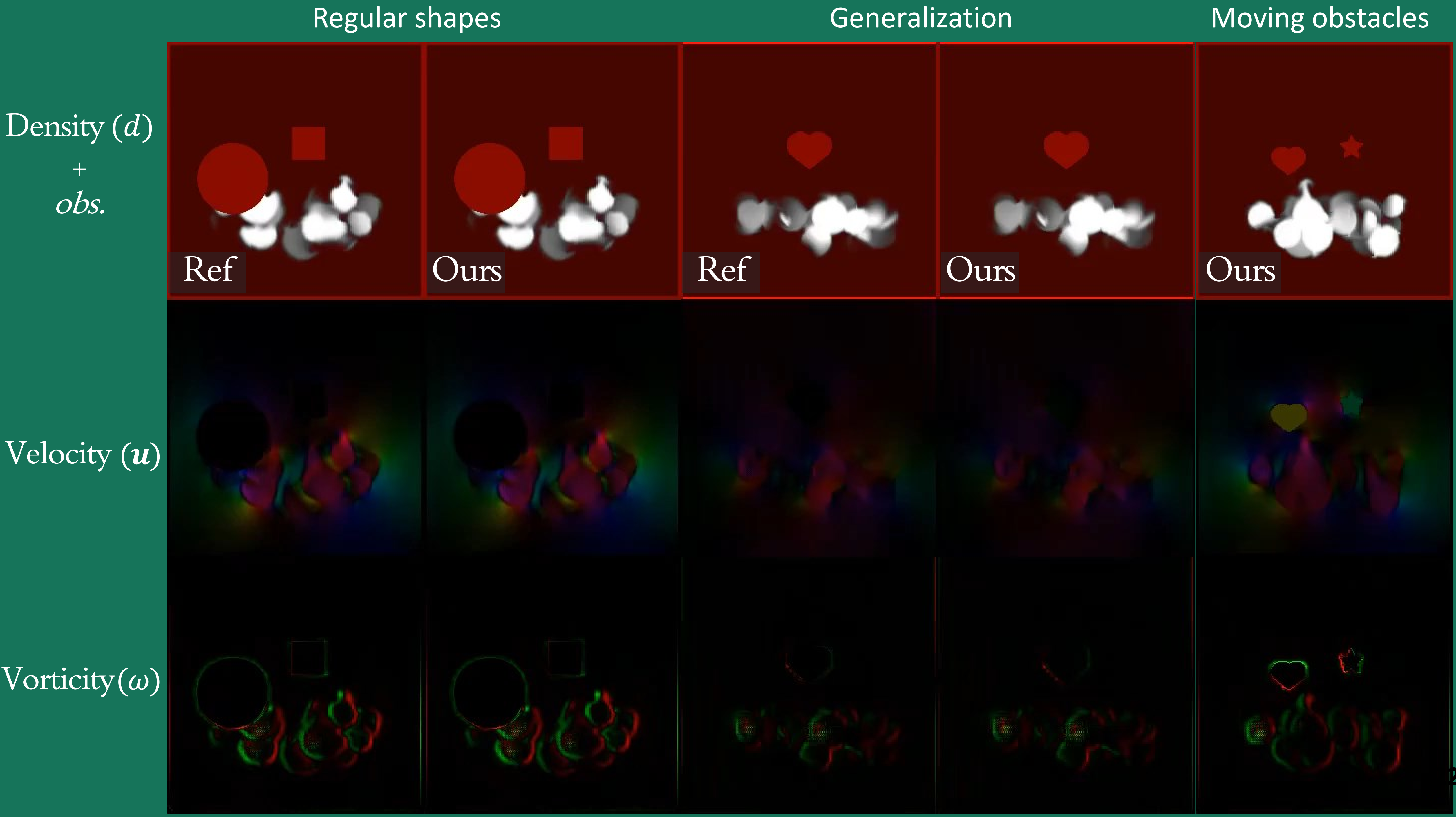

Obstacle design


Velocity difference



Density
d
below training rg.
seen min
unseen interp.
seen max
above training rg.
Velocity
$\boldsymbol{u}$
Vorticity
ω


Buoyancy →
ω × 1.0
ω × 1.5
ω × 2.0
below training rg.
seen min
unseen interp.
seen max
above training rg.



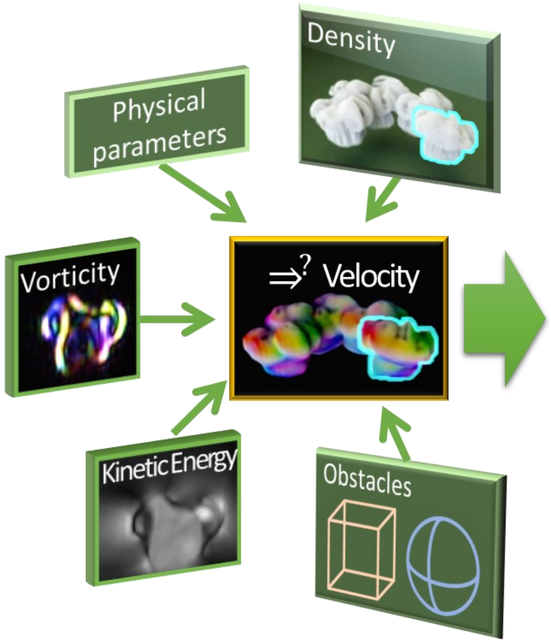
Regular shapes
Generalization
Moving obstacles
Density (d) + *obs.*
Velocity ($\boldsymbol{u}$)
Vorticity (ω)
Ref
Ours
Ref
Ours
Ours

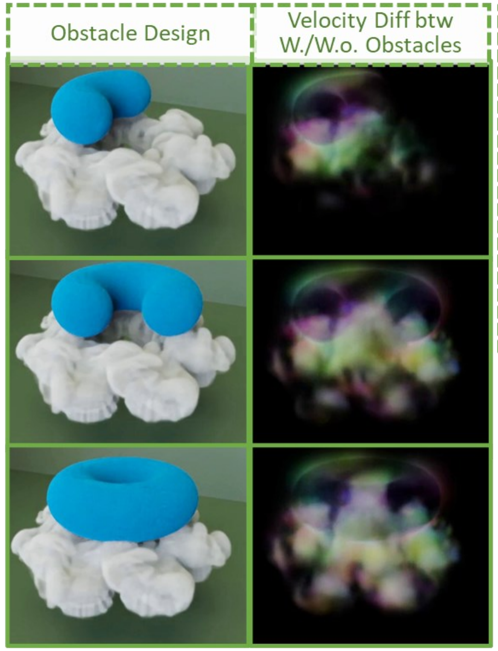

Replay

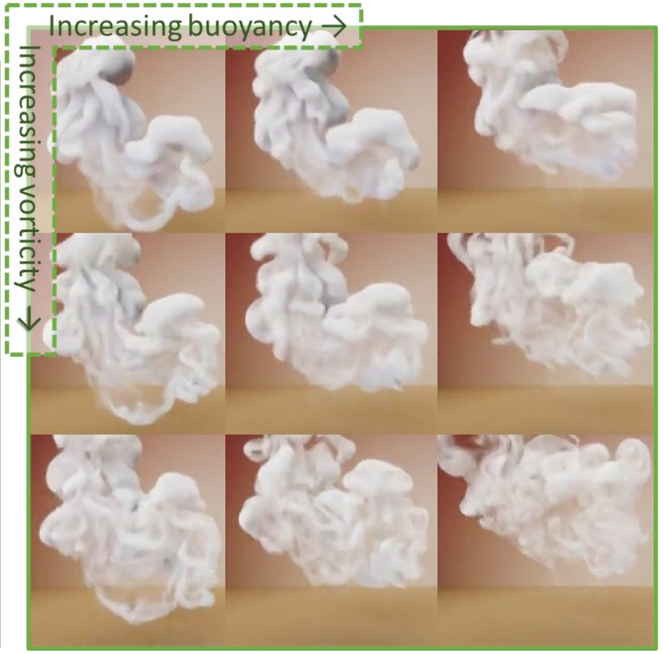
Replay

Conclusions

SIGGRAPH 2021



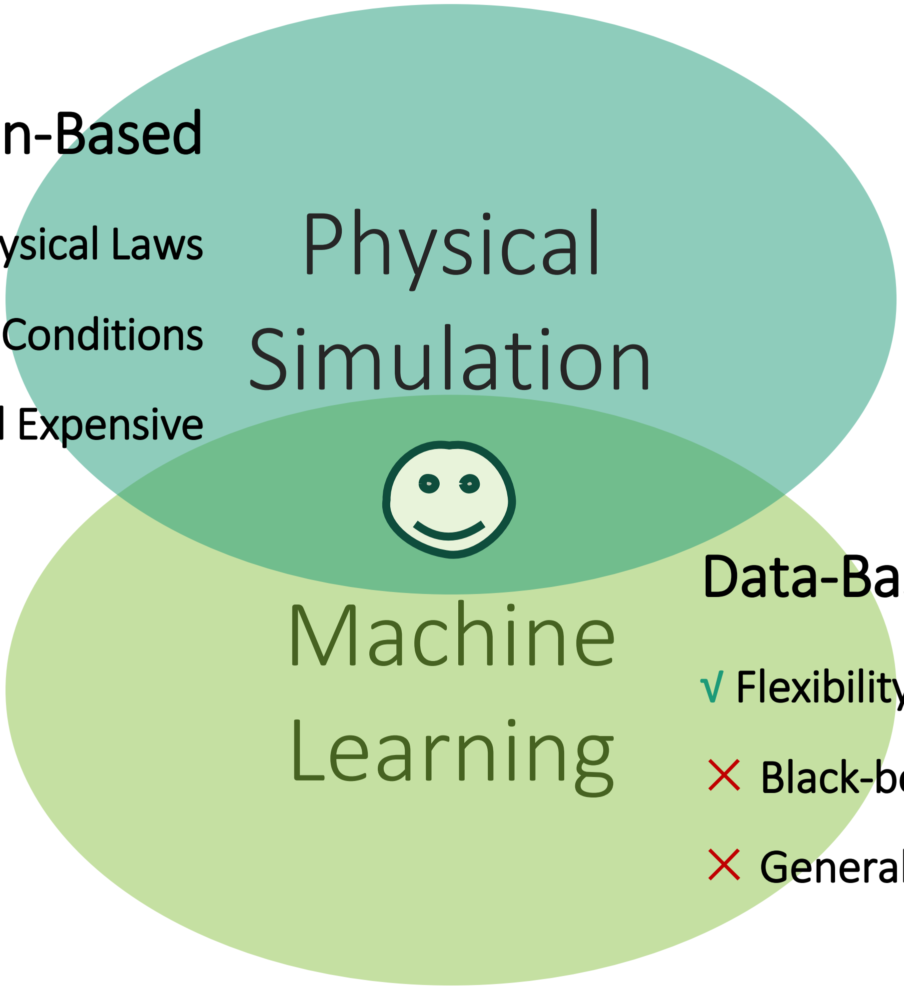




Code & More information:
github.com/RachelCmy/den2vel

- Contributions:
  - Simulations with single-density input
  - Multiple controls simultaneously
  - Highly-sensitive cyclic GAN
  - Strong generalizability
- Limitations:
  - Historical information, 3D resolution, …
- Future directions:
  - Visual controls (streamlines, captures)

Conclusions



Equation-Based

√ Obey Physical Laws

✕ Rely on Initial Conditions

✕ Computational Expensive

Data-Based

√ Flexibility, Efficiency

✕ Black-box Models w.o. Physical Understanding

✕ Generalization