Convolutional Neural Network Based CFD Simulation

Kok Xuan Lun Nigel
U1520907G

Supervised by A/P Li Hua

13 May 2019

Presentation Outline

1. Introduction and Motivations
2. Methodology
3. Results and Discussions
4. Conclusion

Introduction

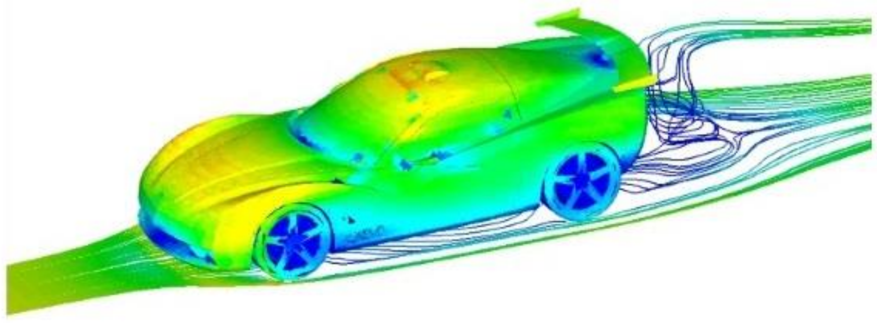
Thermodynamic systems

Mechanical systems

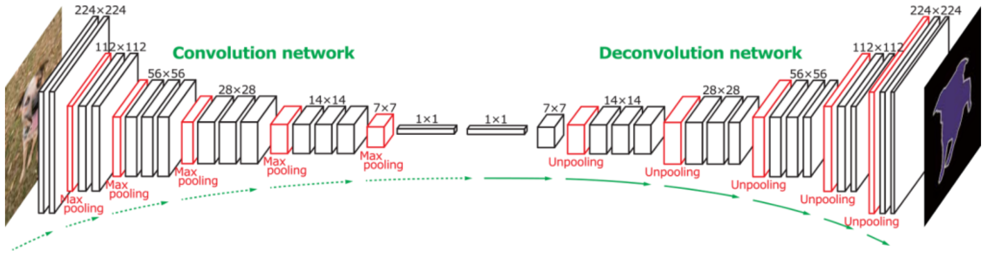
Electrical systems

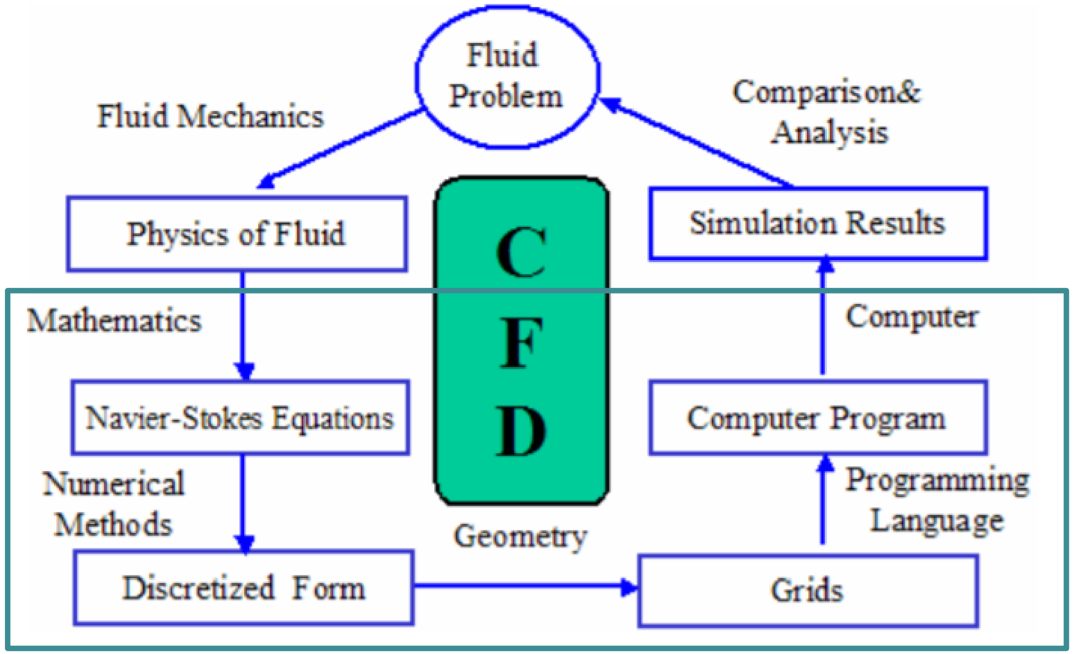
Aerodynamic systems

Motivation



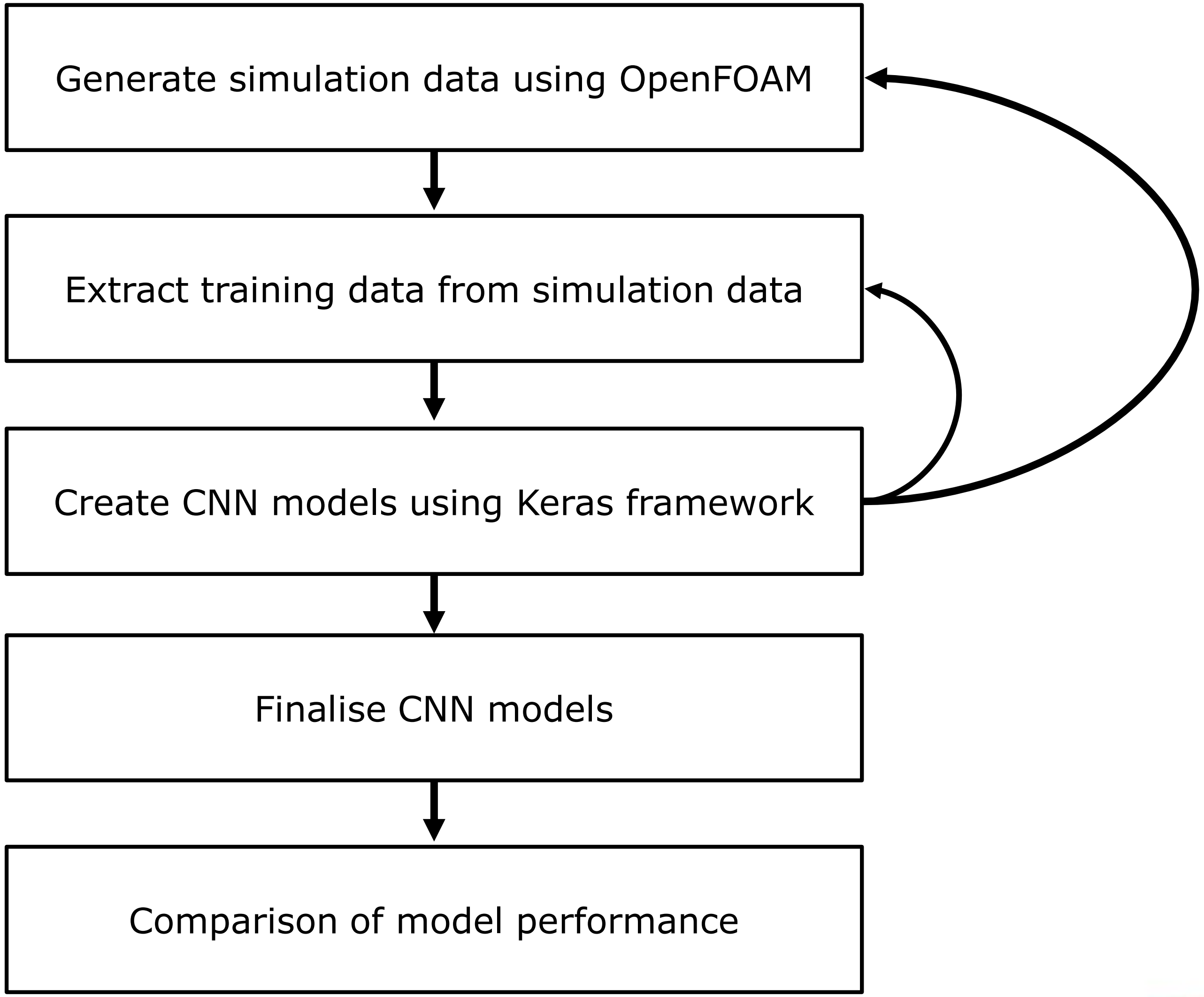
+





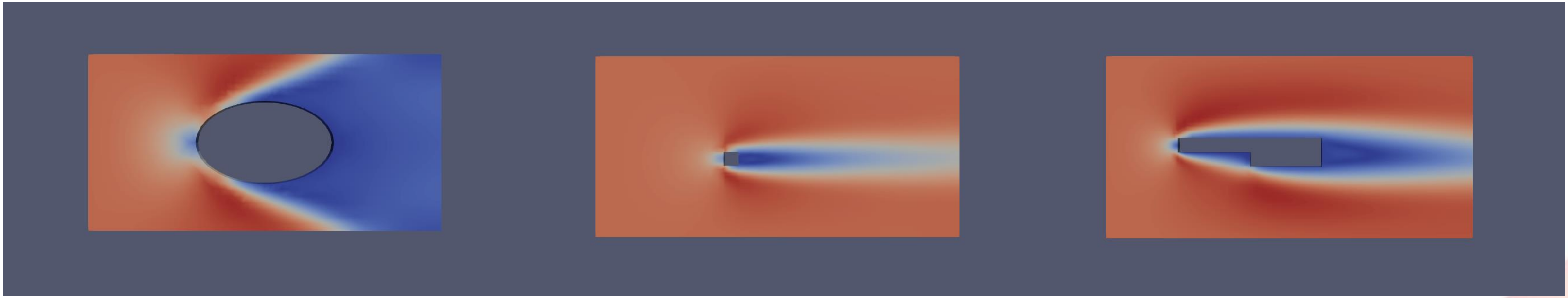
- Improve design prototyping processes:
  - Faster predictions
  - Lower computational costs
- Enabled due to advancements in computer technology

Methodology

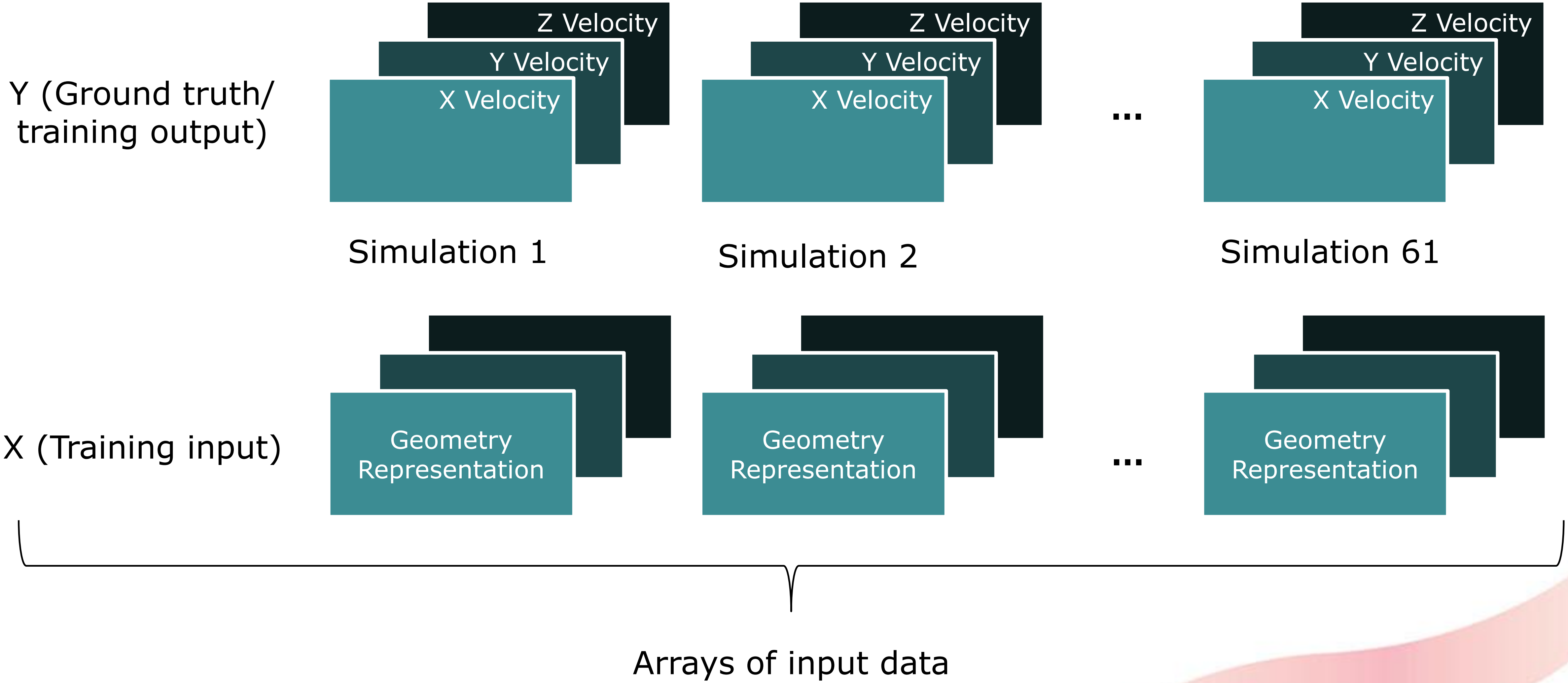


Methodology: Simulation Data

- 64 data sets (61 for network training, 3 for testing)
- Simple geometries (3-4 different base geometries with variations in lengths, in different locations in space)
- 2D simulation with 1 initial condition (0.3 m/s entry velocity)
- Simulated up to 500s



Methodology: Input Data



Methodology: Input Data

X (Training input)

Array of shape (61, 128, 256, 3)

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	-1	-1	-1	0	0	0
0	0	0	0	-1	-1	-1	0	0	0
0	0	0	0	-1	-1	-1	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

Methodology: Input Data

x	y	z	U_0	U_1	U_2
0	0	0	0.3	-0.0004	0
0	0.1	0	0.3	-0.00024	0
0	0.2	0	0.3	-8.08E-05	0
0	0.3	0	0.3	0	0
0	0.4	0	0.3	0	0
0	0.5	0	0.3	0	0
0	1.5	0	0.3	0	0
0	1.6	0	0.3	0	0
0	1.7	0	0.3	0	0

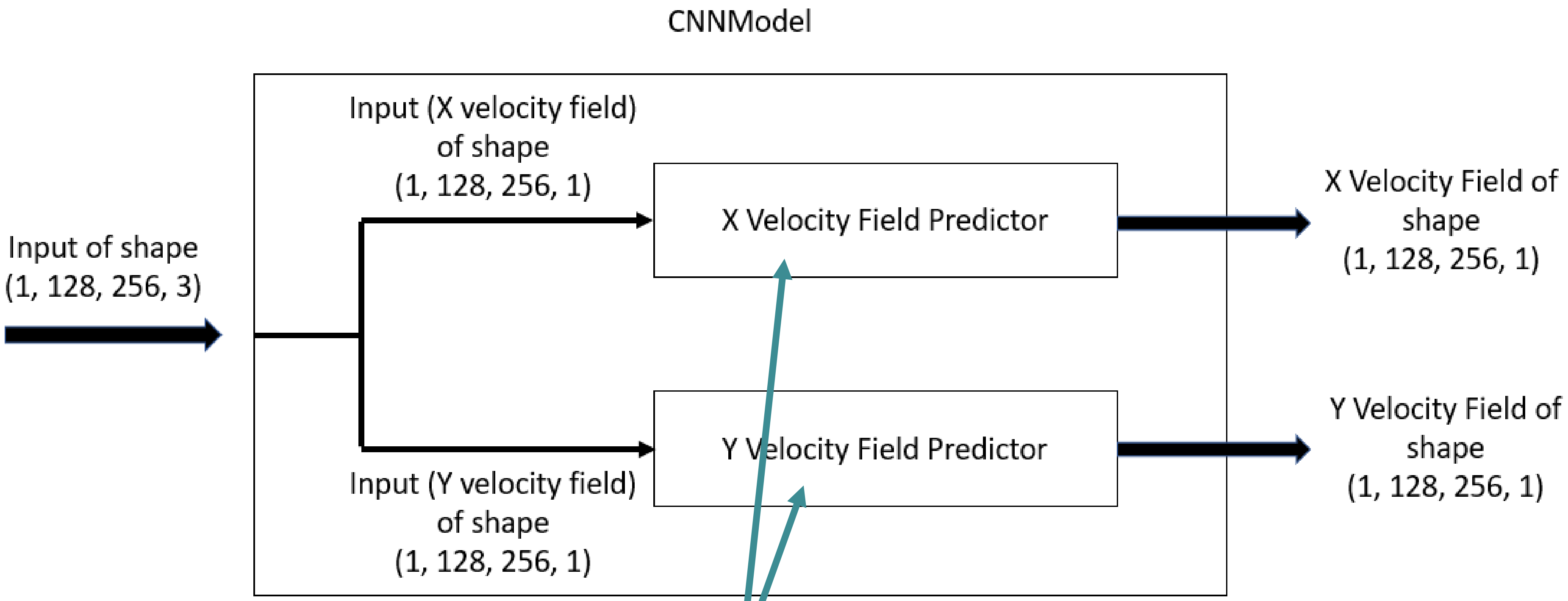
Y (Ground Truth)
Array of shape (61, 128, 256, 3) for (Number of simulation cases, Y coordinates, X coordinates, Velocity fields)

Velocity field data
X, Y and Z coordinates with corresponding X, Y and Z components of velocity

x	y	z	p
0	0	0	0.000121
0	0.1	0	0.000266
0	0.2	0	0.000411
0	0.3	0	0.000581
0	0.4	0	0.000775
0	0.5	0	0.00097
0	0.6	0	0.001165
0	0.7	0	0.00136
0	0.8	0	0.001556
0	0.9	0	0.001752
0	1	0	0.001948
0	1.1	0	0.002146
0	1.2	0	0.002344
0	1.3	0	0.002543
0	1.4	0	0.002743
0	1.5	0	0.002943
0	1.6	0	0.003144
0	1.7	0	0.003345

Pressure field data
X, Y and Z coordinates with corresponding pressure values

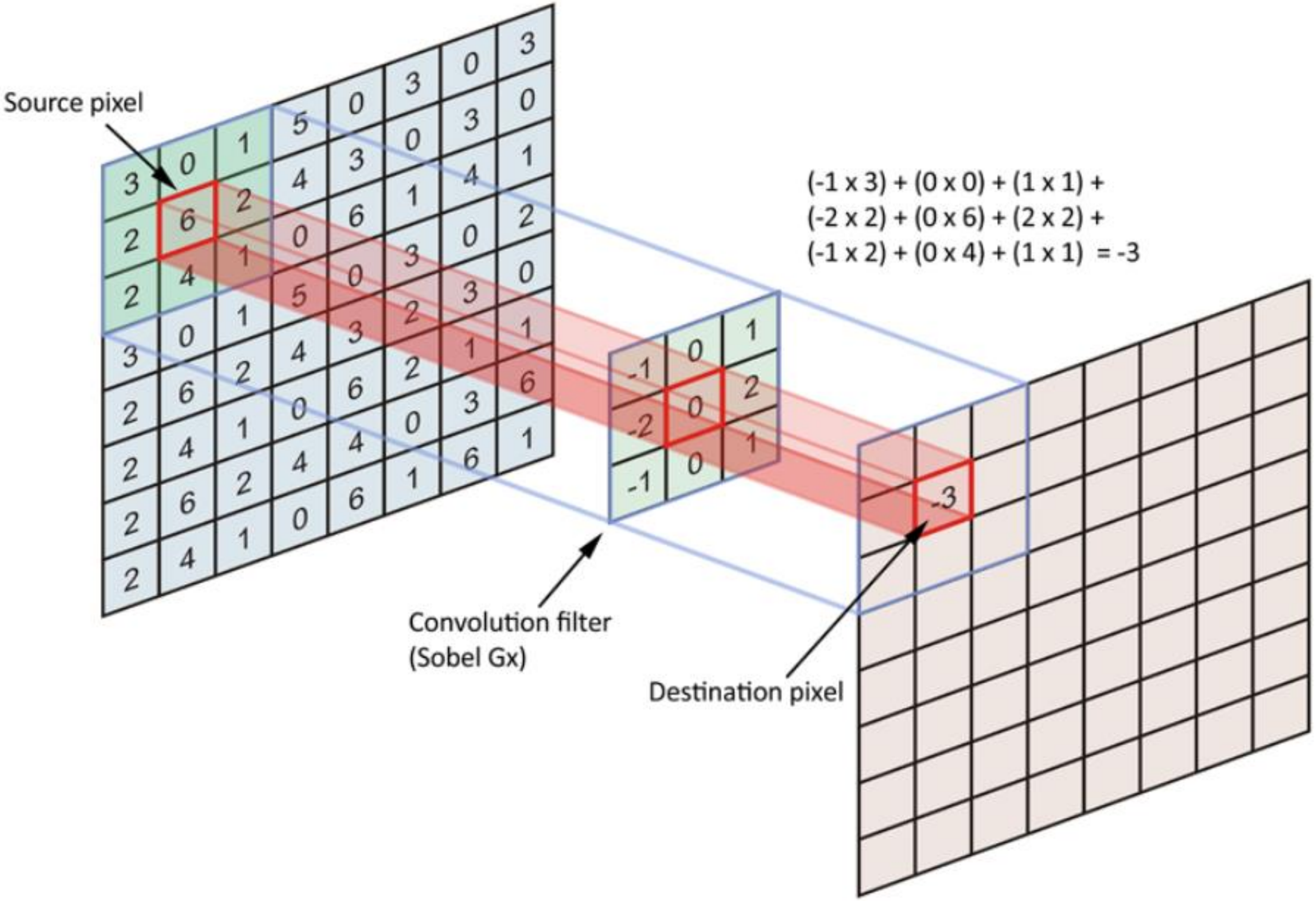
Methodology: Model



3 Variations

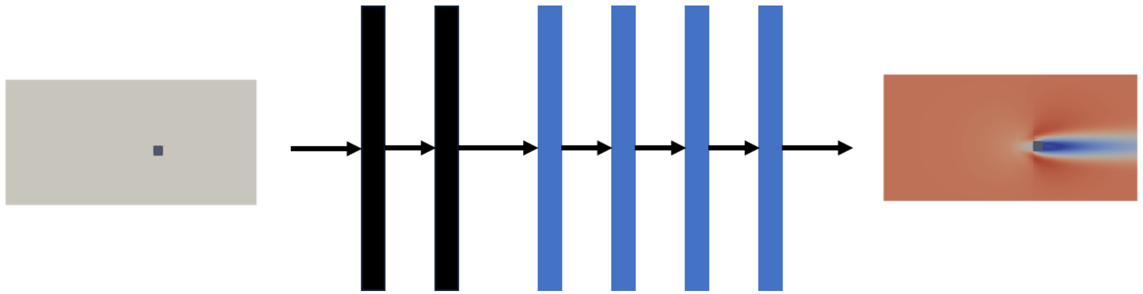
1. Fully convolutional network
2. Strided convolutional network
3. Strided convolutional network with 1 fully connected layer

Methodology: Model

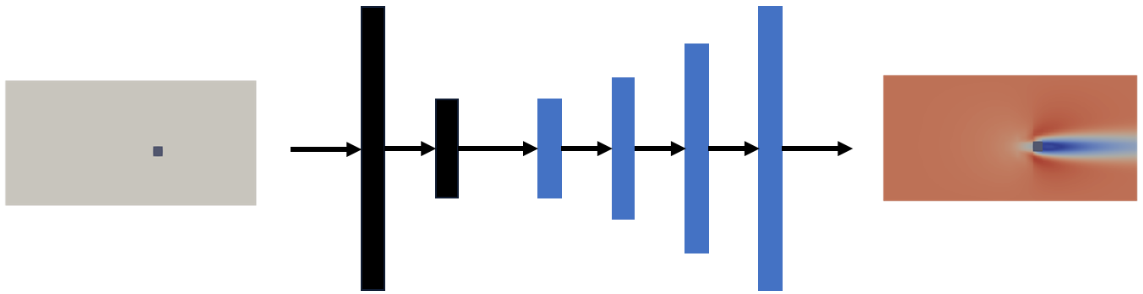


Methodology: Model

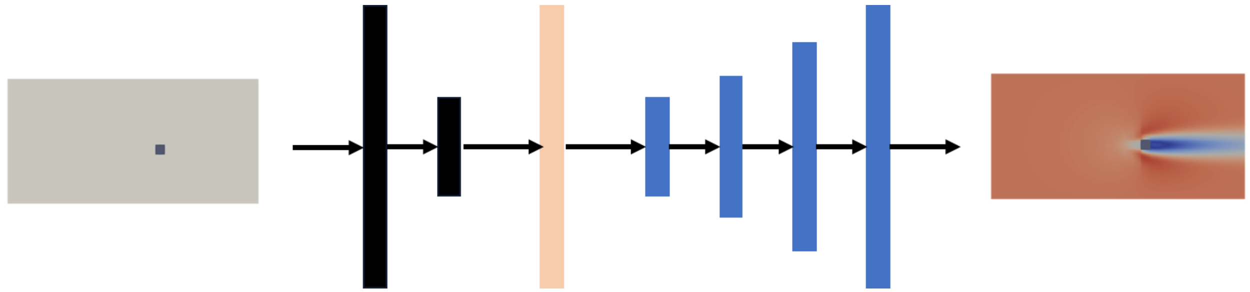
Fully convolutional network
(CNNModel 1)



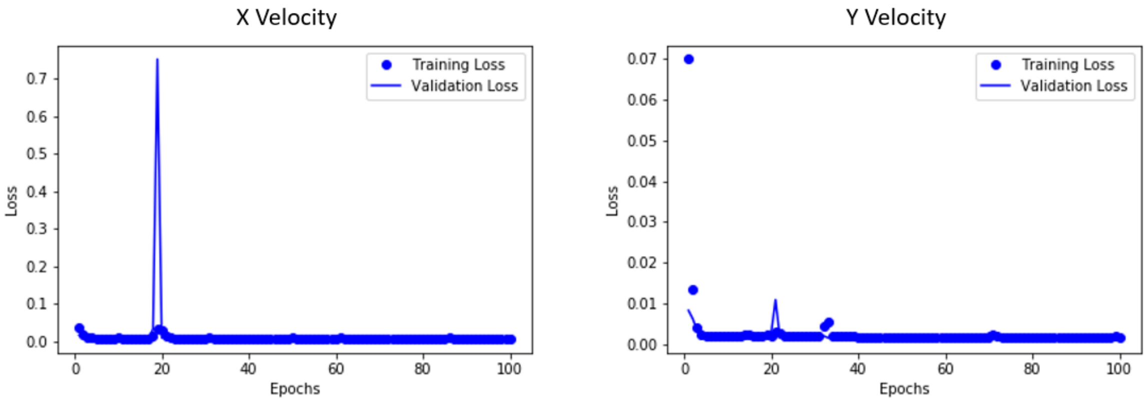
Strided convolutional
network
(CNNModel 2)



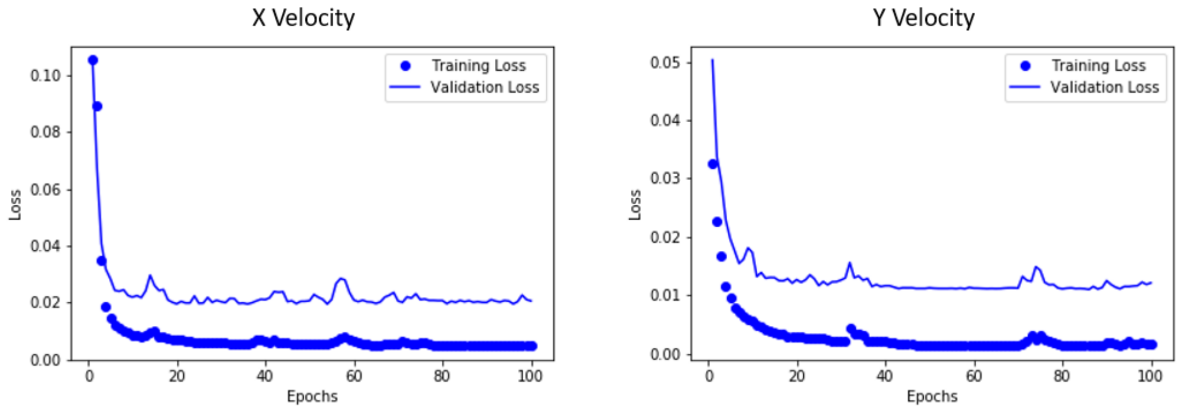
Strided convolutional
network with 1 fully
connected layer
(CNNModel 3)



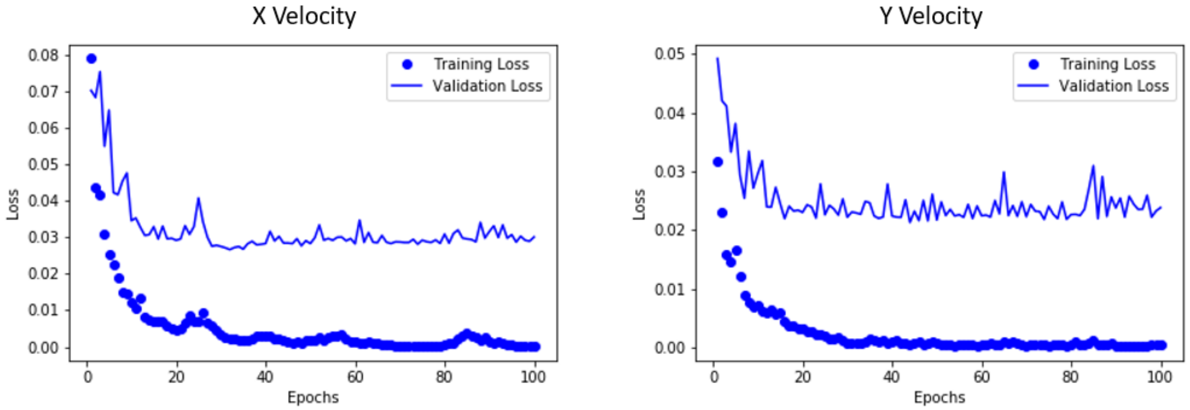
Results and Discussion



CNNModel 1 (Full convolution)



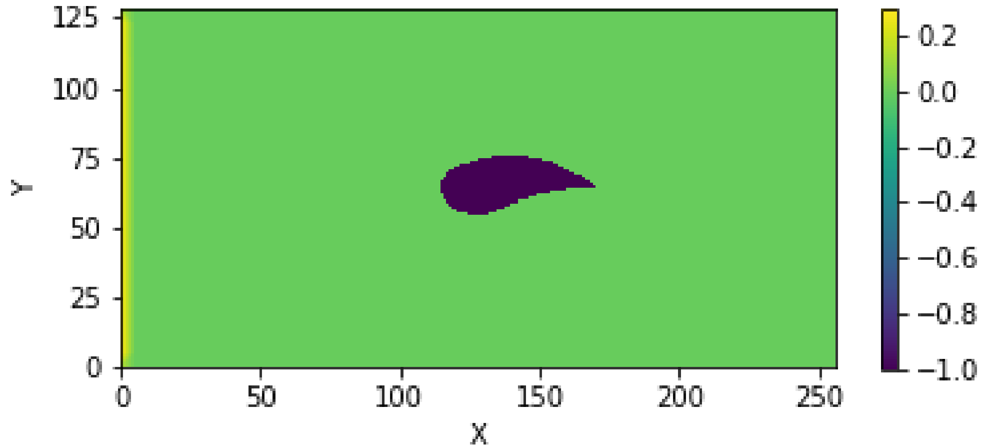
CNNModel 2 (Strided convolution)



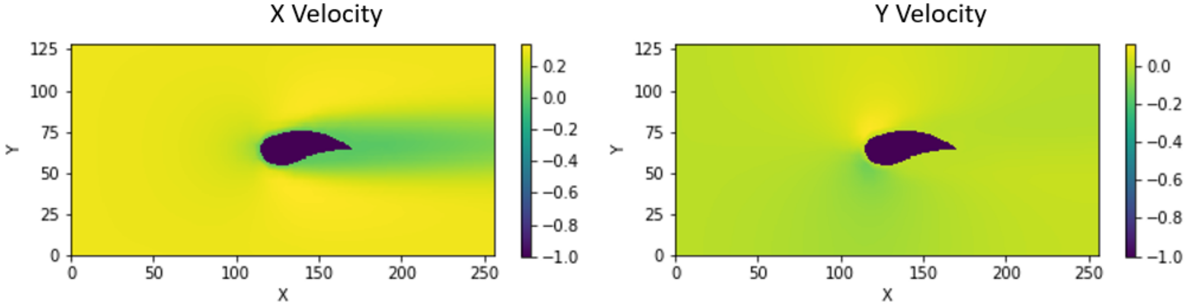
CNNModel 3 (Strided convolution with FC layer)

Decent training losses but significantly higher validation losses ➔ Model's high complexity

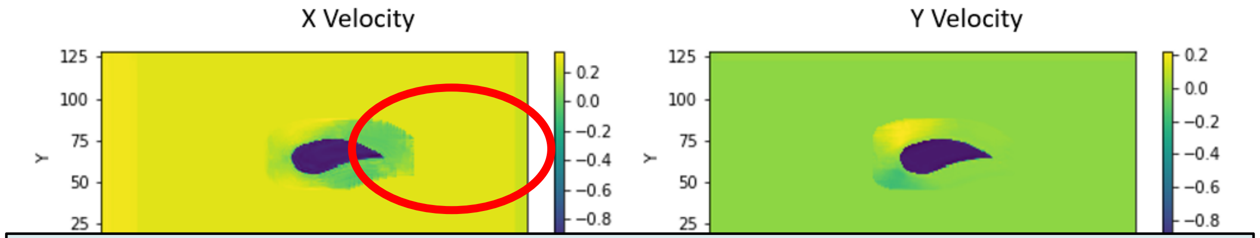
Results and Discussion



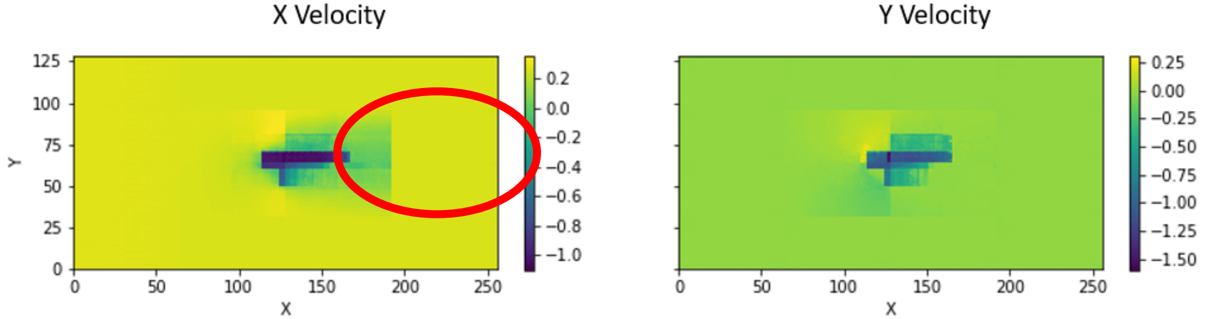
Sample 1 - Input



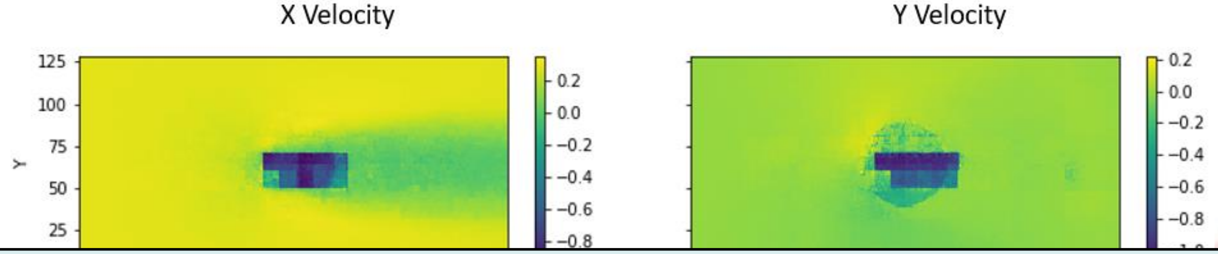
Sample 1 – Ground truth



Strong ability to capture obstacle geometry but weaker ability to predict velocity fields further from obstacle

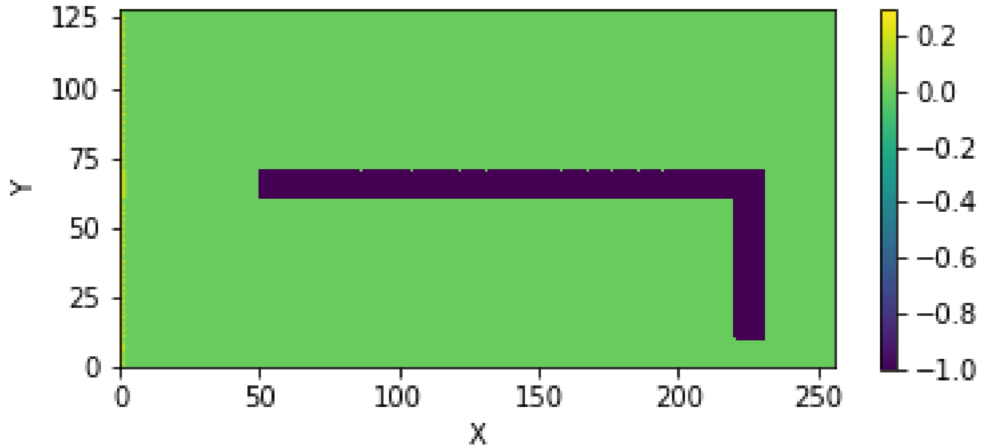


Sample 1 – CNNModel 2

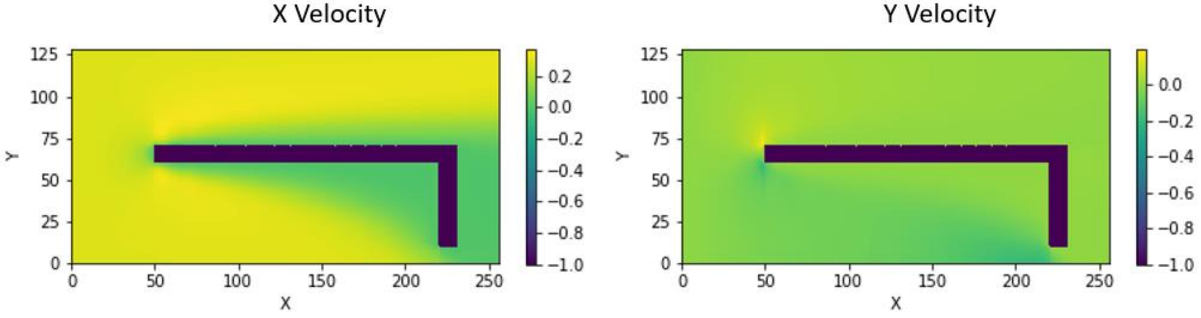


Smooth replication of velocity field in boundary and wake region but poor ability to capture obstacle geometry

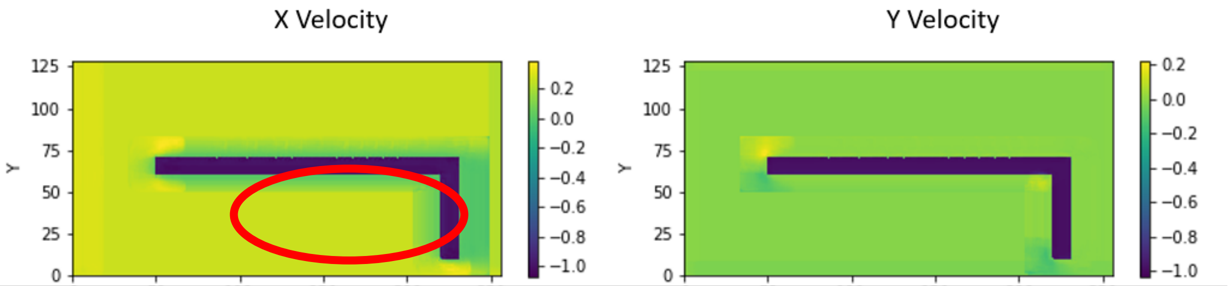
Results and Discussion



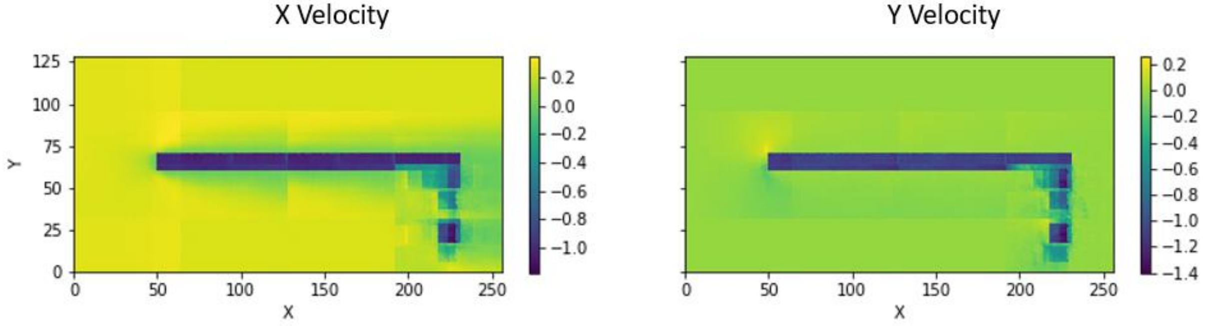
Sample 2 - Input



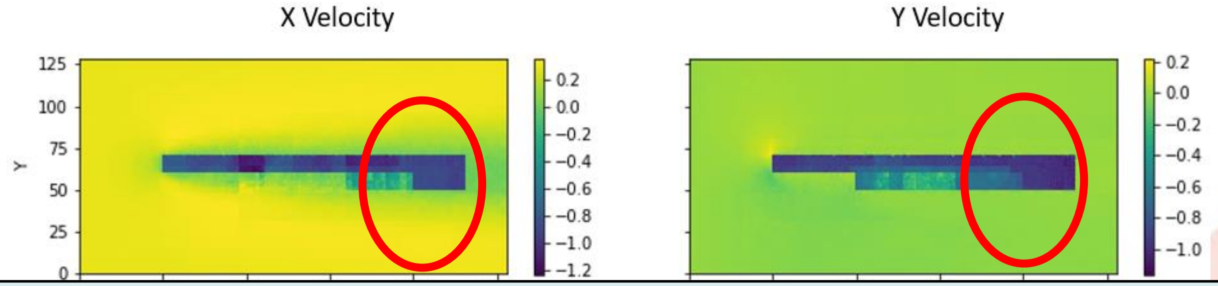
Sample 2 – Ground truth



Strong ability to capture obstacle geometry but weaker ability to predict velocity fields further from obstacle

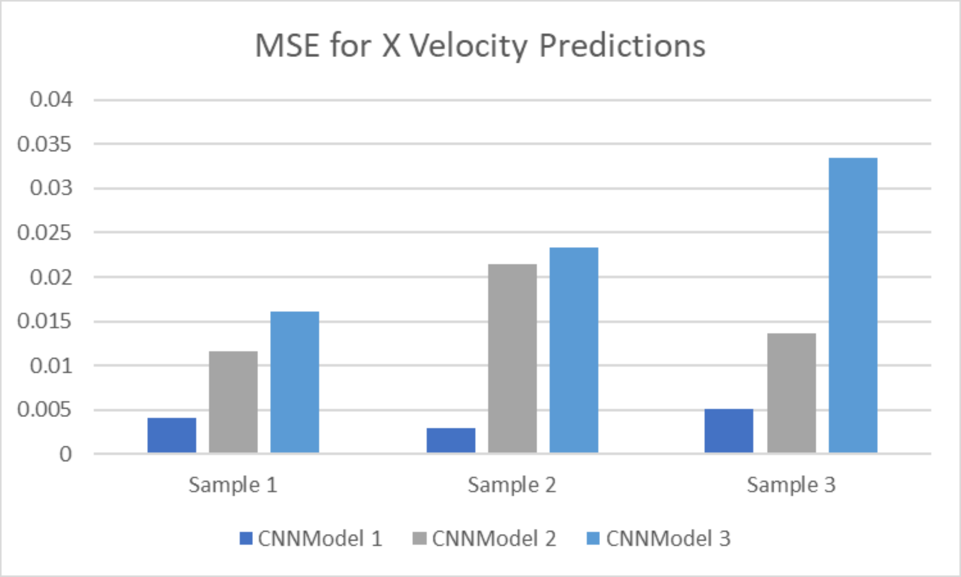


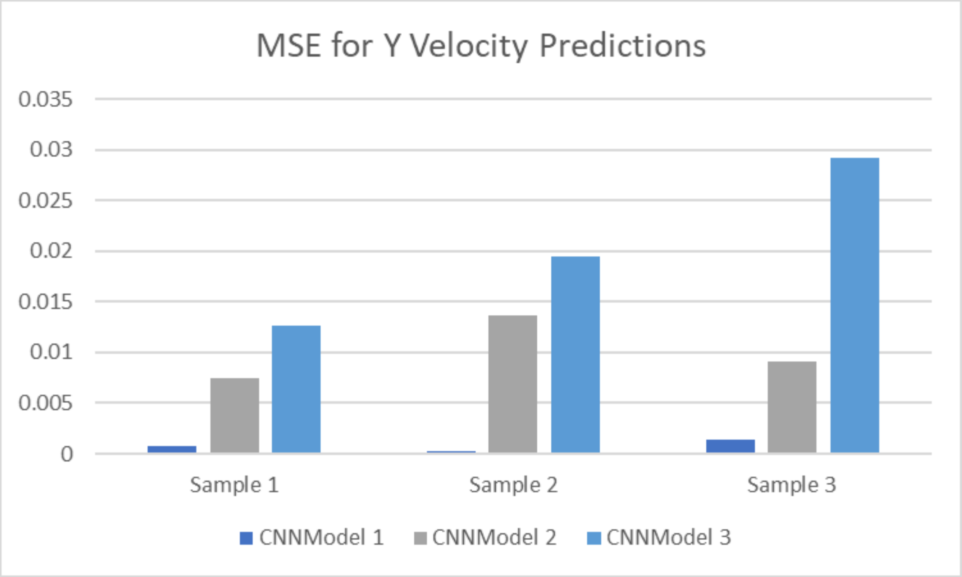
Sample 2 – CNNModel 2



Failure to accurately predict obstacle geometry but boundary layer is still smooth

Results and Discussion





- Fully convolutional network has the lowest MSE
  - Better network training
  - Possibly attributable to more accurate obstacle geometry prediction

However, fully the convolutional network lacks important traits provided by strided convolutions with fully connected layers

  - Though better in performance, potential may be limited

Results and Discussion

- Inverse relationship between ability to capture obstacle geometry and generating smooth prediction of velocity fields in boundary and wake region

- Spatial information is lost as data is downsampled

- Accurate velocity field predictions contingent on model’s ability to accurately output obstacle geometry

- Fully connected layers are important to bridging the relationship between cells that are further away from each other

Results and Discussion

Advantages of machine learning in CFD:

- Time and cost savings
- Potentially accurate results within scope of the problem

Disadvantages of machine learning in CFD:

- Requires a large amount of simulation data
  - Different initial conditions, fluid properties, boundary conditions
- Model specific to flow characteristics of simulation data
  - Laminar flow simulation data does not contain intrinsic information of turbulent flow simulation data ➔ Separate models possibly necessary
- Implicit assumptions: Steady state exists, laminar flow etc.

Future Work

- **Characterisation studies**
    - Permutations of model hyperparameters

- **Modification of network architecture**
    - Addition of skipped connections
    - Addition or removal of layers (and changes to their hyperparameters)

- **Data set coverage**
    - Different initial conditions
    - Different boundary conditions (2D vs 3D problems)
    - Laminar and turbulent flows

Conclusion

- Convolutional neural networks showed strong ability in dealing with problems where spatial information is important (beyond its initial use for classification problems)
  - Machine learning has evolved to handle a wider variety of problems

- Machine learning may potentially find their way into computational engineering fields (structural analysis, fluid analysis, design optimisations)
  - Not in the near future due to reliability concerns (safety etc.)

- Potential to change how engineering processes are managed

End of Presentation