Probing Particle Impingement in Boilers Using Computational Fluid Dynamics (CFD) with ANSYS Fluent

Steve Yang and Hyuna Kwon
Supervisor : Dr. Bryan Wong
University of California, Riverside
5/24/2021

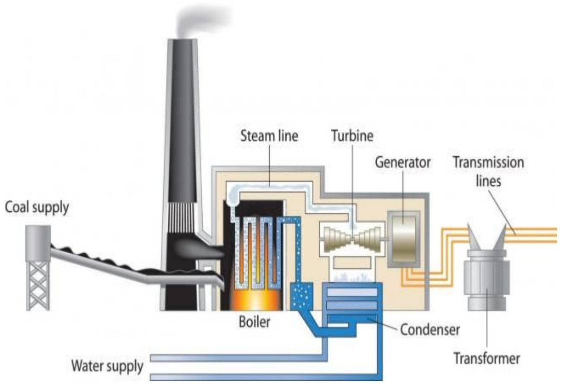
Outline

Predicting solid particle erosion

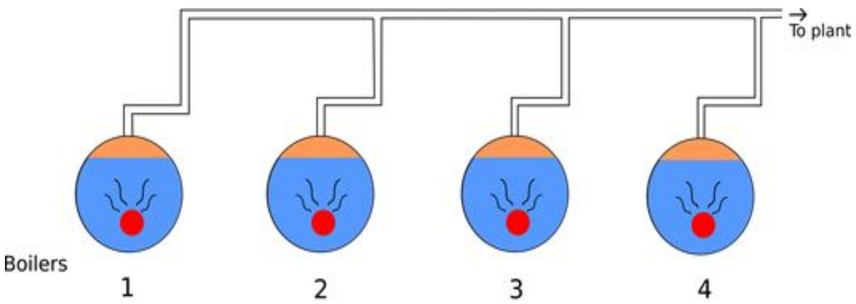
- Solid particle erosion :
  material damage caused by solid particles that impinge a surface

- Resulting in inefficient processes and financial instability.
- Boiler header is a pipeline that transfers steam from boiler heads to plant

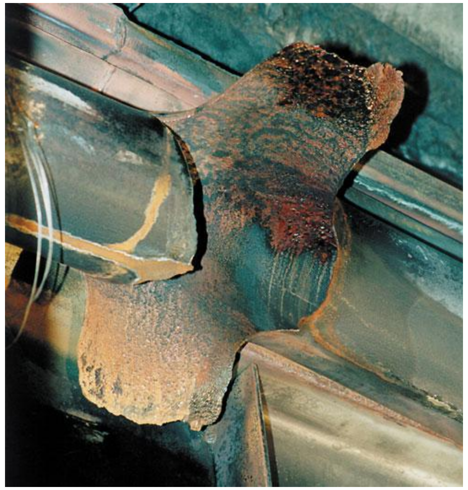
Coal plant



Schematic drawing of coal plant



Boiler header



Damaged pipe

However, erosion is a complex phenomenon

Predicting solid particle erosion

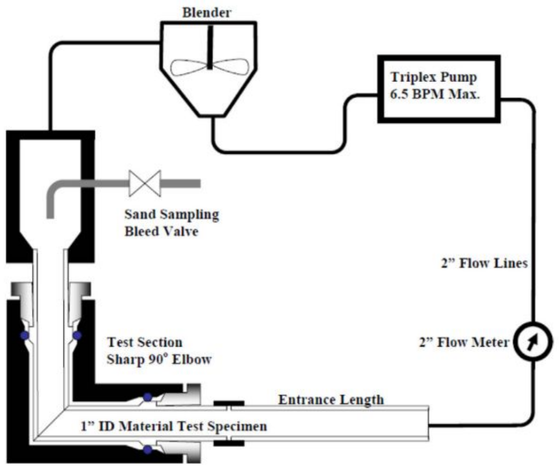
Previous attempts to predict erosion rate

Extensive experiments and field studies

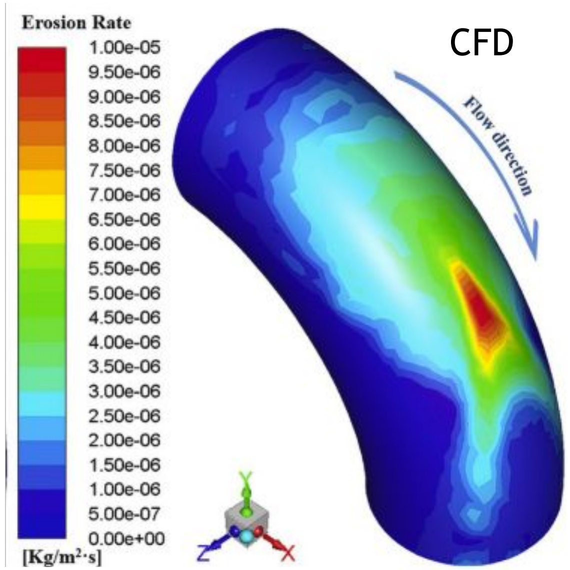
Computational fluid dynamics (CFD)

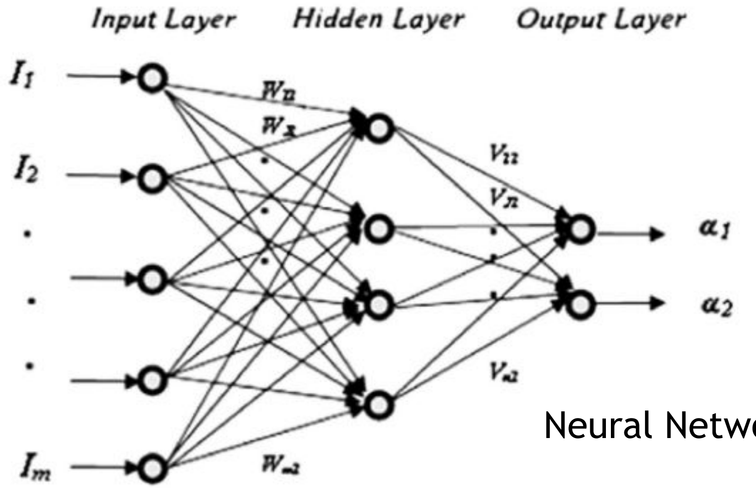
Machine learning approach

Experiments



CFD



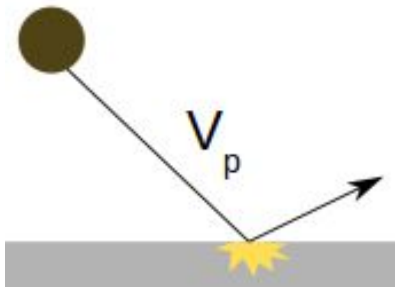


Neural Network

Factors affecting erosion rate

(1) Particle velocity

$ER \propto V_p^{\,n}$



V_p : particle velocity

Various values of n have been proposed

Finnie : n = 2
Oka : n = function of hardness

(2) Impingement angle

(3) Particle properties (hardness, size, shape, etc.)

Oka and McLaury developed erosion models incorporating particle properties

General formulation under DPM

$$ER = \sum_{p=1}^{N_{traject}} \frac{\dot{m}_p C(d_p) f(\alpha) v_p^n}{A_{face}}$$

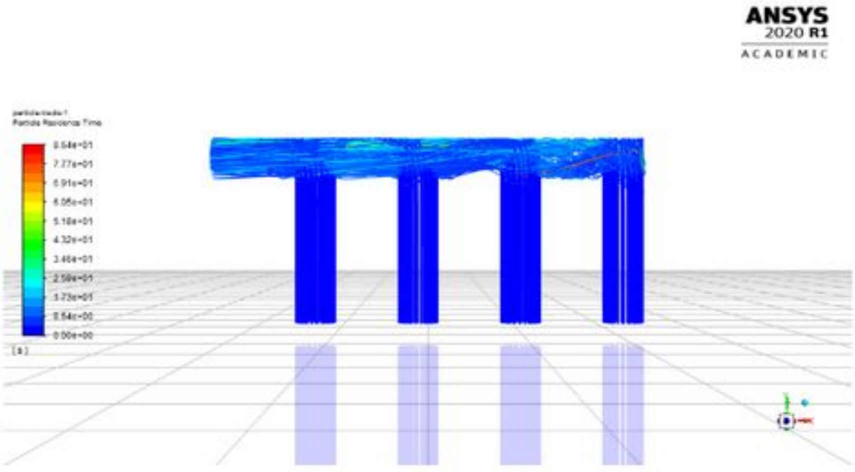
m_p	:	***Mass flow rate of the particles***
$f(\alpha)$	:	***Impact angle*** function
v_p	:	***Particle impact velocity***
n	:	Velocity exponent
$C(d_p)$	:	***Particle diameter*** function

CFD-based erosion modeling

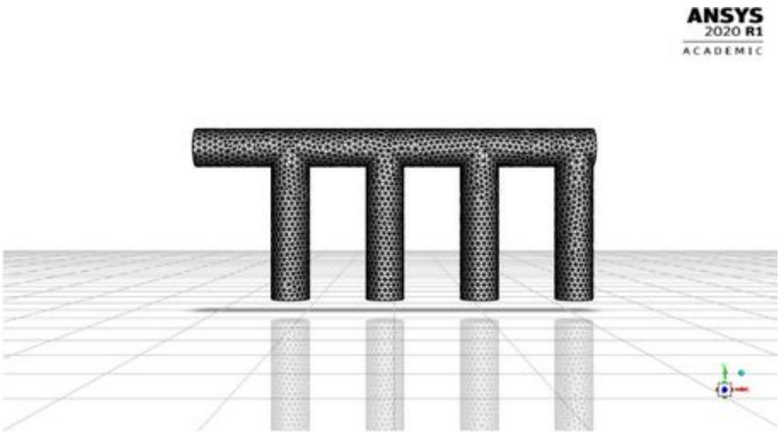
(1) Pros & Cons

- Low-cost, fast solution
- Complex task because it involves various phenomena that must be considered

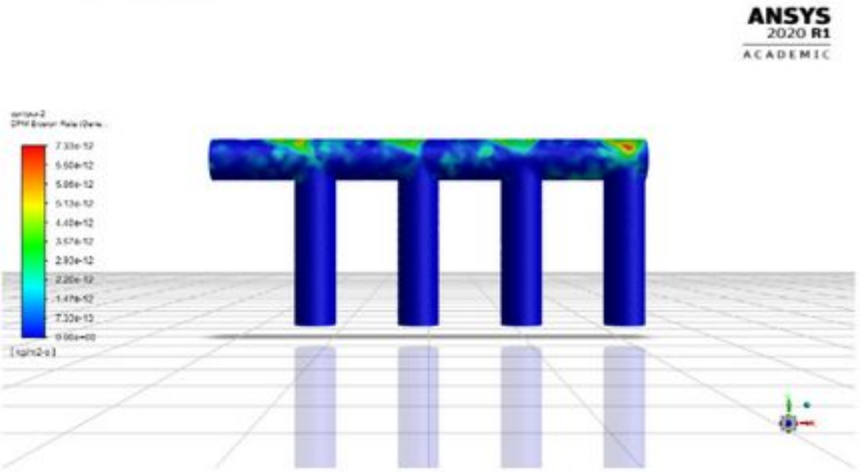
(2) Procedure



Fluid flow modeling →



Discrete particle modeling →



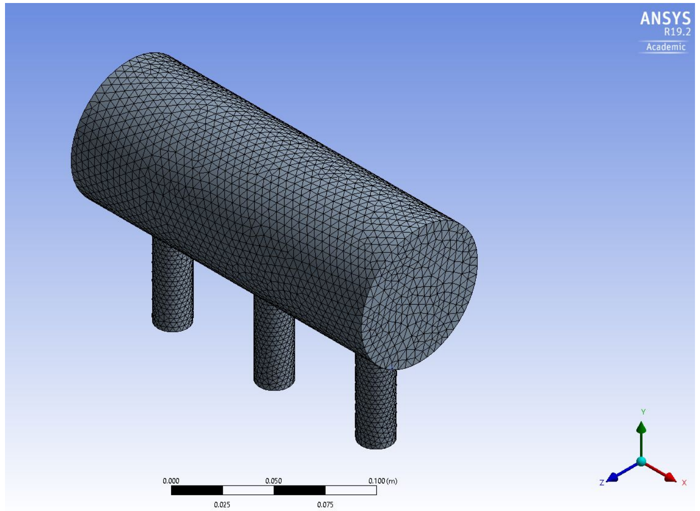
Erosion modeling

ANSYS Fluent

- Used ANSYS Fluent v 19.2 to generate data
- Finite-volume CFD code that solves the Navier-Stokes equation to model fluid flow

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla)\mathbf{v} = \frac{\mathbf{F}}{\rho} - \frac{1}{\rho}\nabla p$$

$$\frac{\partial}{\partial t}(\rho u_i) + \frac{\partial}{\partial x_j}(\rho u_i u_j) = -\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}[\underbrace{\mu \frac{\partial u_i}{\partial x_j}}_{\text{Viscosity}} - \rho \frac{\partial u_i}{\partial x_j} - \underbrace{\rho \overline{u_i' u_j'}]}_{\substack{\text{Turbulence} \\ (k\text{-}\varepsilon \text{ model})}} + \underbrace{\rho g_i}_{\text{Gravity}} + F_i$$



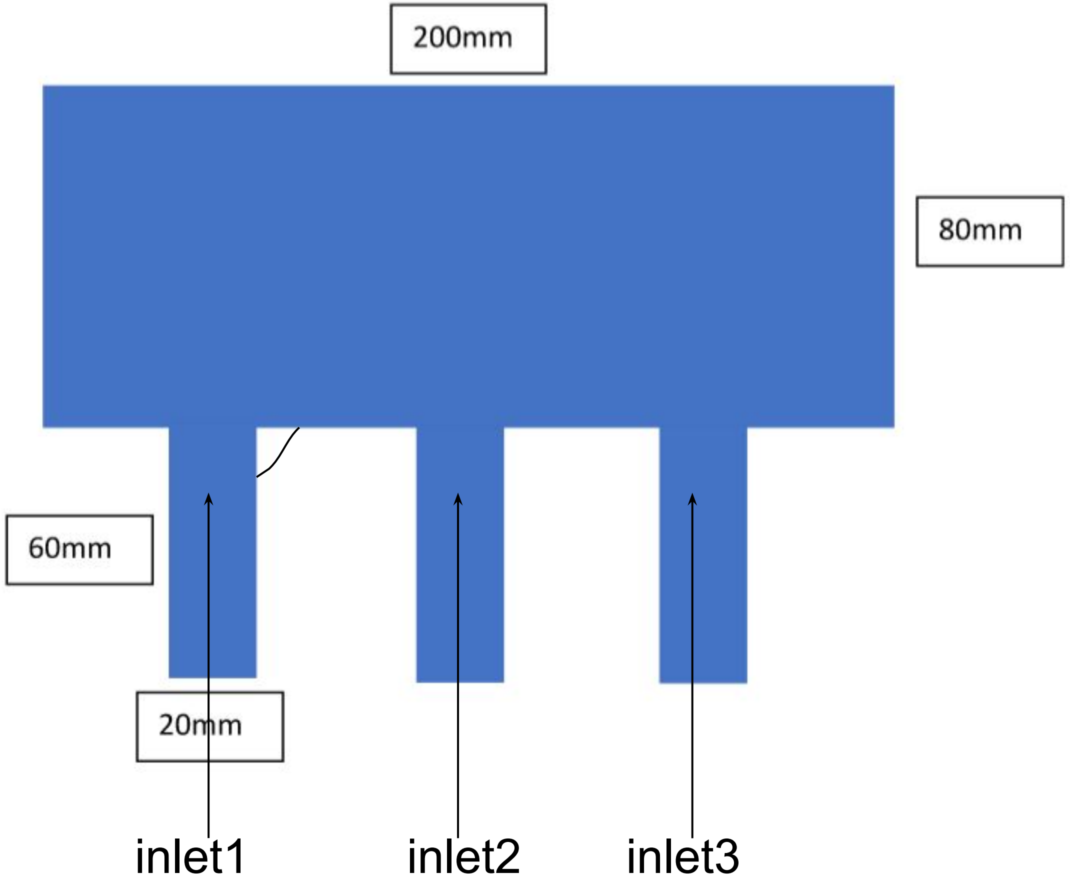
- ANSYS automated meshing was used to discretize the domain

Setup variables in ANSYS Fluent

- Average Mesh area: 7.8985e-003 m^2

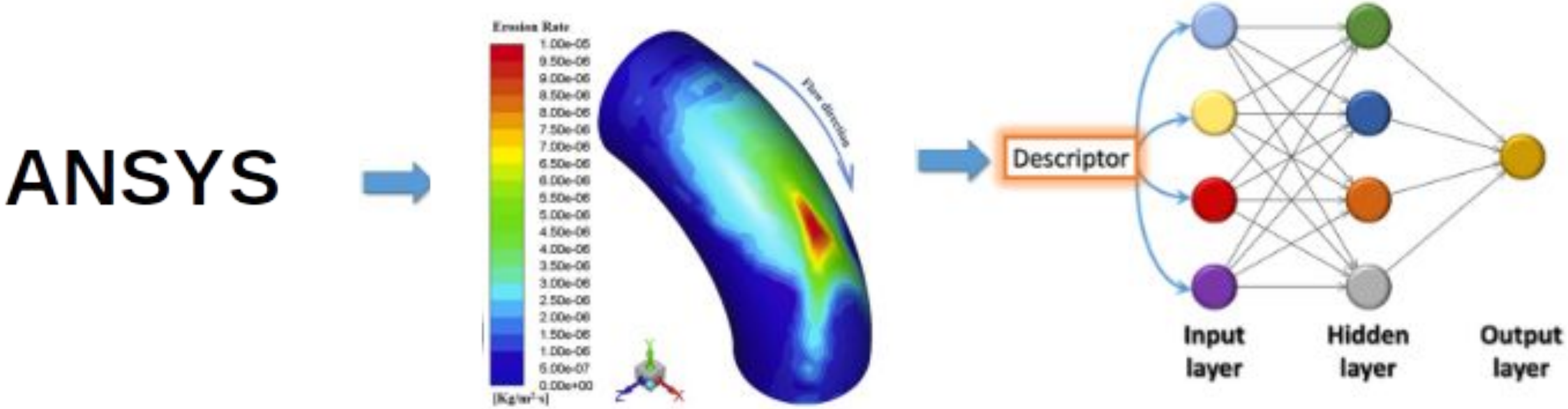
- **Variables**
- Viscous: SST k-omega
- Fluids: air / water vapor
- Soild: steel
- Inert Particle type: Anthracite
- Particle diameter: 2e-05 m
- Particle velocity magnitude: 45m/s
- Particle flow rate 0.00454 kg/s
- **Inlet conditions (inlet)**
- Inlet pressure: 70000 Pa
- Inlet velocity: 12.45 m/s
- **Small inlet conditions (inlet_1, inlet_2, inlet_3)**
- Small inlet pressure: 200000pa
- Small inlet pressure: 45m/s

- # of iterations: 200



Outline of this research

- **Hybrid approach of combining CFD and machine learning models** to build a predictive modeling pipeline for erosion rates.
- ANSYS Fluent is used to simulate fluid flow and particle transport on a boiler header.
- Exploratory data analysis was performed on particle trajectories and parameters to identify their significance in erosion rate predictions.
- CFD output is used as input to develop CNN model.



Variables for the CFD simulation

System setup variables
Particle size
Particle velocity
Particle flow rate
Pressure of the main inlet
Velocity at the main inlet
Pressure of the branched inlet
Velocity at the branched inlet

Dependent variables
Particle position vectors
Particle velocity vectors
Erosion rate
Erosion location

Artificial neural network

- Too many variables which are directly correlated with the result. so automatization of the variable settings and simulation running procedure is necessary.
- The more data available, the higher performing model can be obtained.

Convolutional Neural Network(CNN) model

- CNN: a convolutional neural network (CNN, or ConvNet) is a class of deep neural networks, most commonly applied to analyzing visual imagery by summarizing the presence of features in an input data.

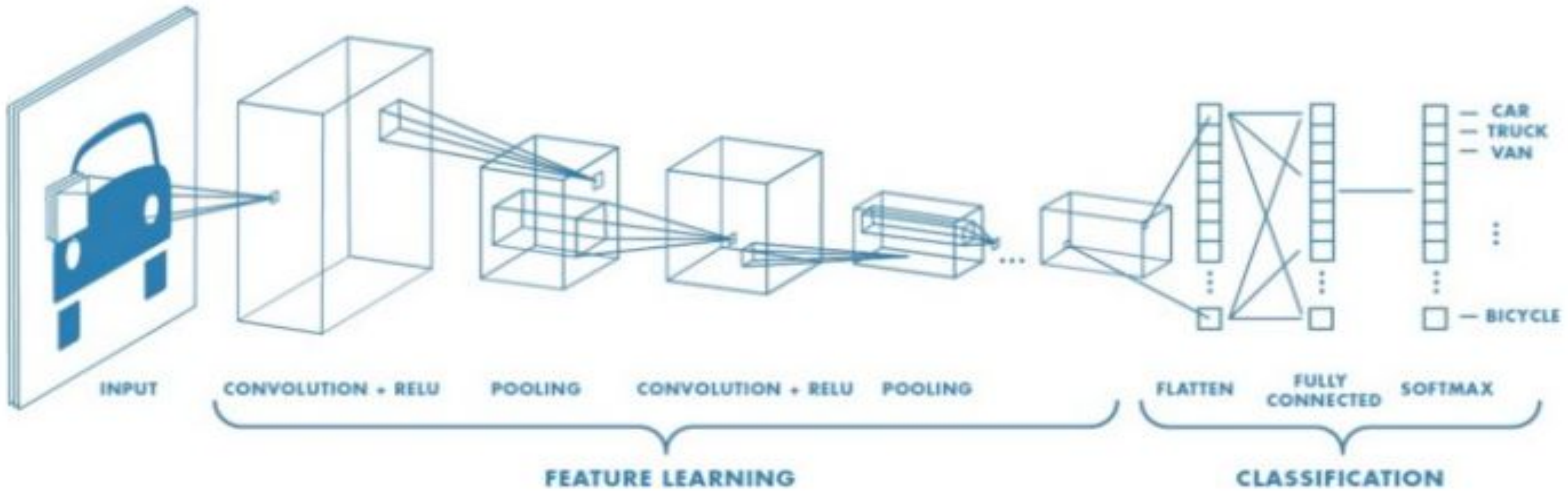


Figure 2 : Neural network with many convolutional layers

Ref: Towards Data Science Inc

Comparison of CNN for Video recognition and CFD

The input shape(data dimension) of the video recognition with CNN is very similar to the input of the erosion prediction with computational fluid dynamics

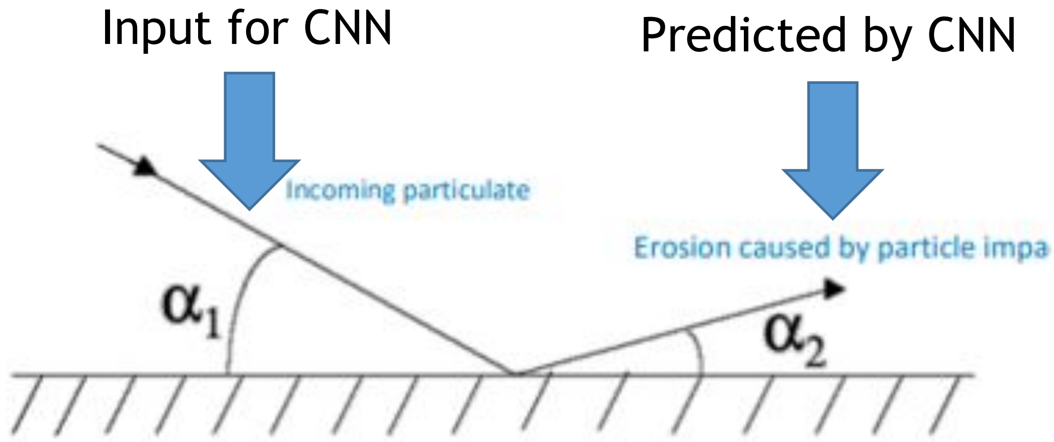
Input shape	Video recognition	CFD
Batch	Number of videos	Number of simulations
In_depth	Number of pictures at a time period	Time steps for particle position and velocity
In_height	Number of pixels	Number of particles per one simulation
In_width	Number of pixels	Cartesian coordinates of particle position and velocity
In_channels	Red, green, blue	1 channel
Function approximation	Classification	Regression

K-fold cross validation

For the CNN model, the whole data should be divided into 'test set' and 'training set' for minimizing the effects of data discrepancies and better understanding of the characteristics of the model.

K-fold cross validation provides a robust estimate of the performance of a model on unseen data. It does this by splitting the training dataset into k subsets and takes turns training models on all subsets except one which is held out, and evaluating model performance on the held out validation dataset.

Particle position and velocity prediction



General formulation under DPM

$$ER = \sum_{p=1}^{N_{traject}} \frac{\dot{m}_p C(d_p) f(\alpha) v_p^n}{A_{face}}$$

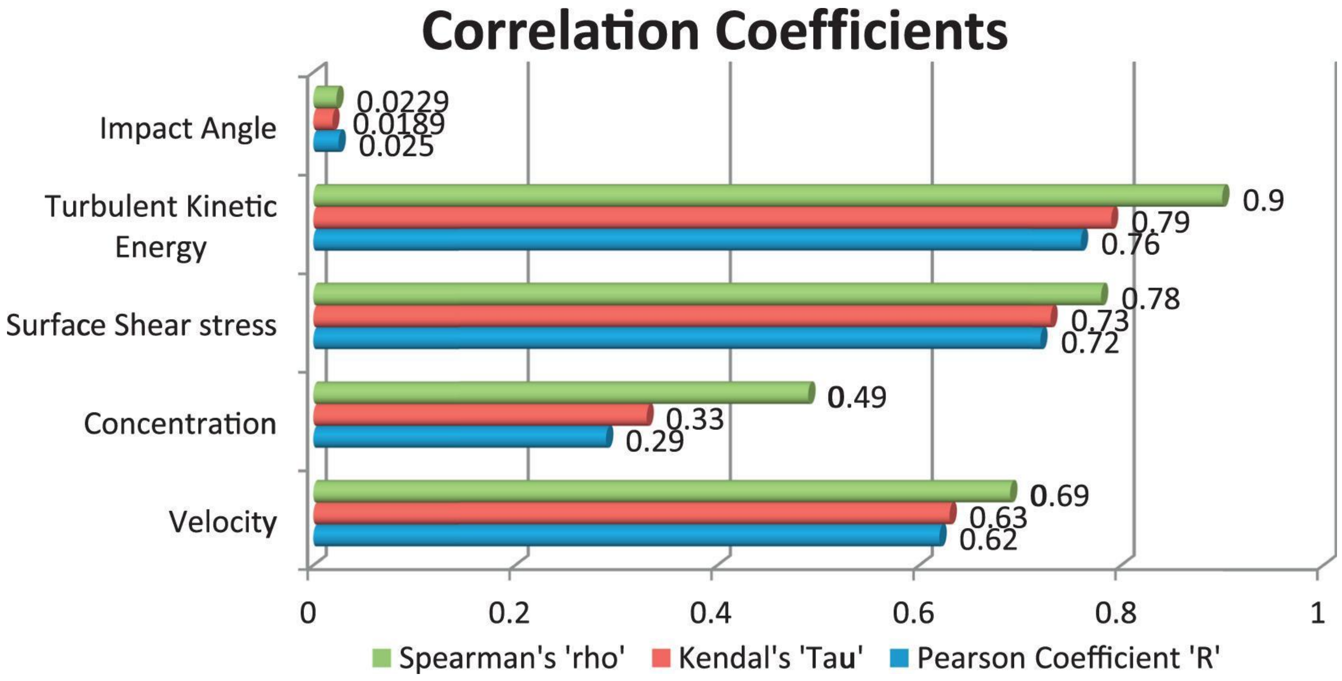
- m_p : *Mass flow rate of the particles*
- $f(\alpha)$: *Impact angle* function
- v_p : *Particle impact velocity*
- n : Velocity exponent
- $C(d_p)$: *Particle diameter* function

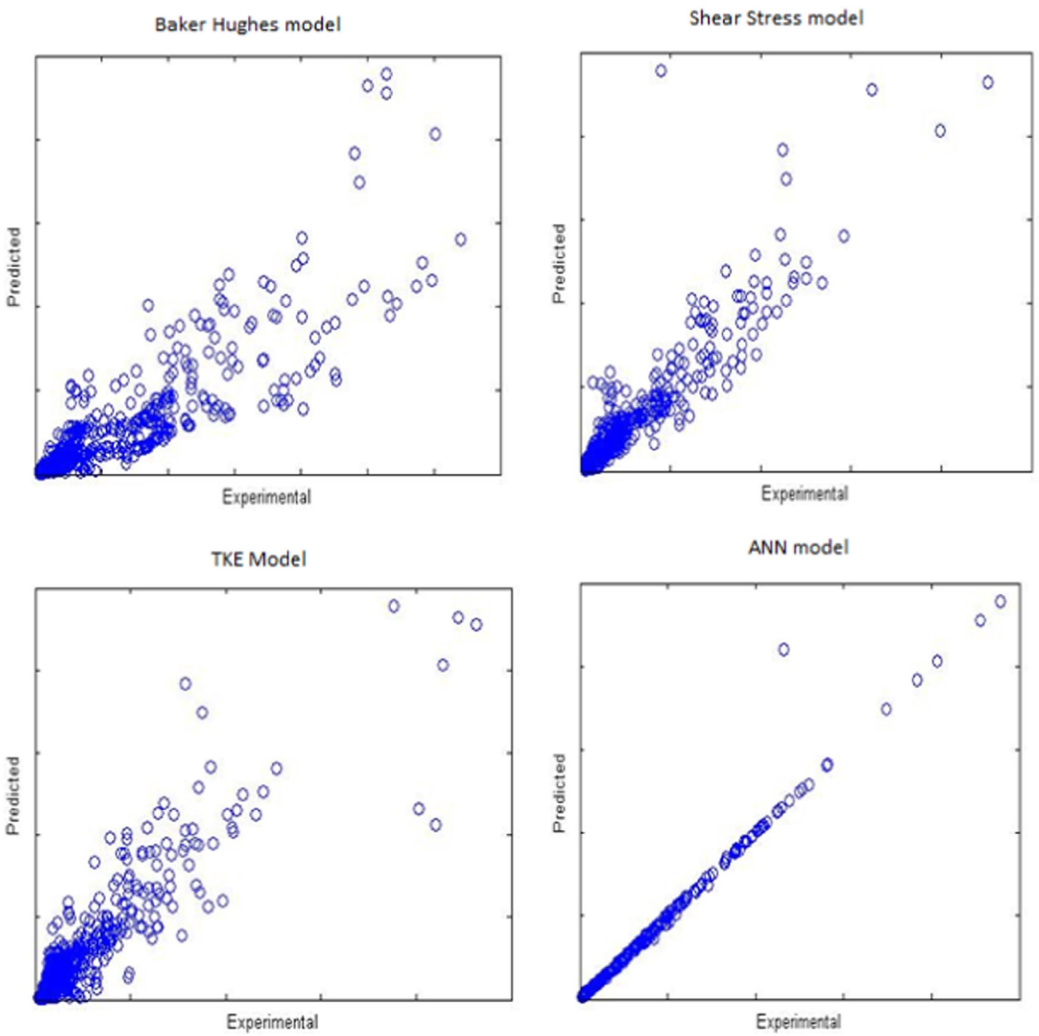
Hypothesis: The erosion rate and the location is predictable if the information of position and velocity after the collision can be predicted by CNN.

The factors on the general formula are deeply related to the trajectory of the particle after the collision.

This will provide an insight of the damage mechanism on the geometry.

Artificial neural network model in CFD





Wear 378 (2017): 198-210.

Research Objective

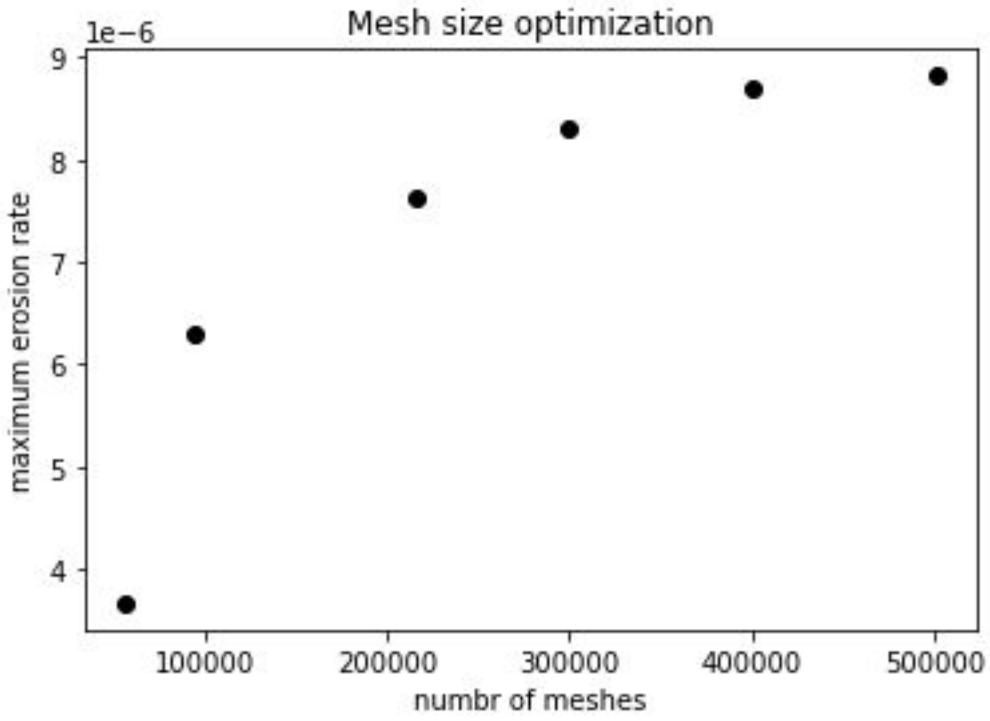
- 1. Providing a new computational analysis (with CNN) to identify and develop insight into the inefficiencies of specific physical processes in existing coal plants

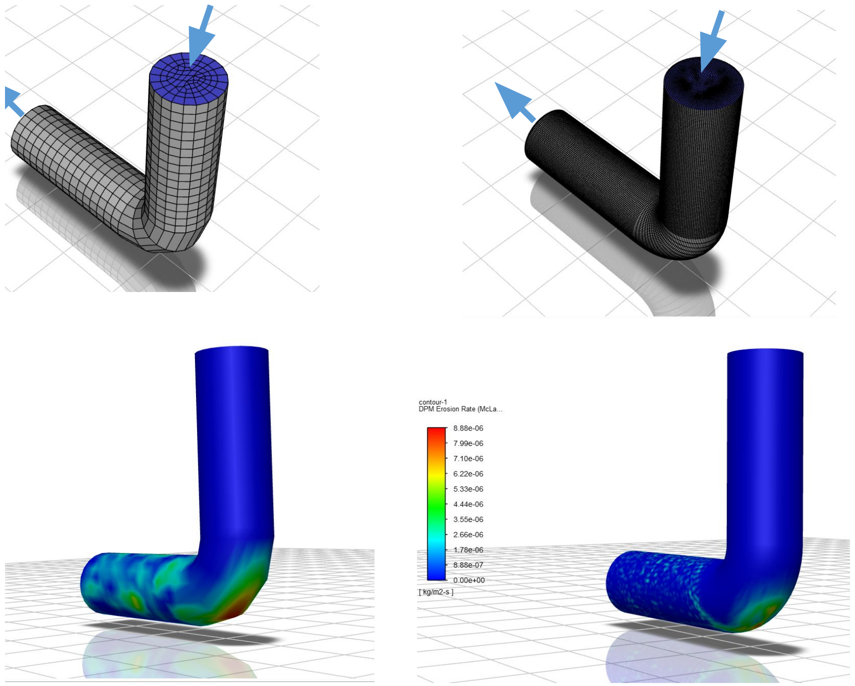
- 2. Predicting Damage Rates of Headers and turbines under different cycling models with artificial neural network model.
  1) Constructing a convolutional neural network model for the particle position and velocity prediction
  2) Developing the convolutional neural network model to predict the erosion location and erosion rate

- 3. Generating numerous dataset with changing system setup variables on Ansys fluent and creating multiple input CNN for high performing erosion prediction model

Preliminary results

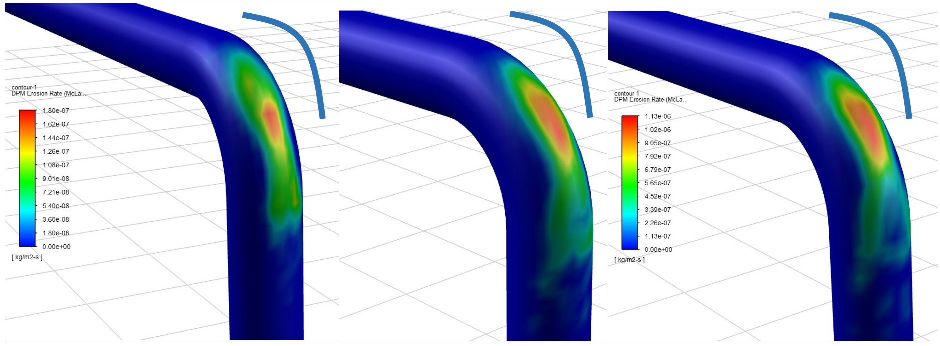
Mesh density optimization



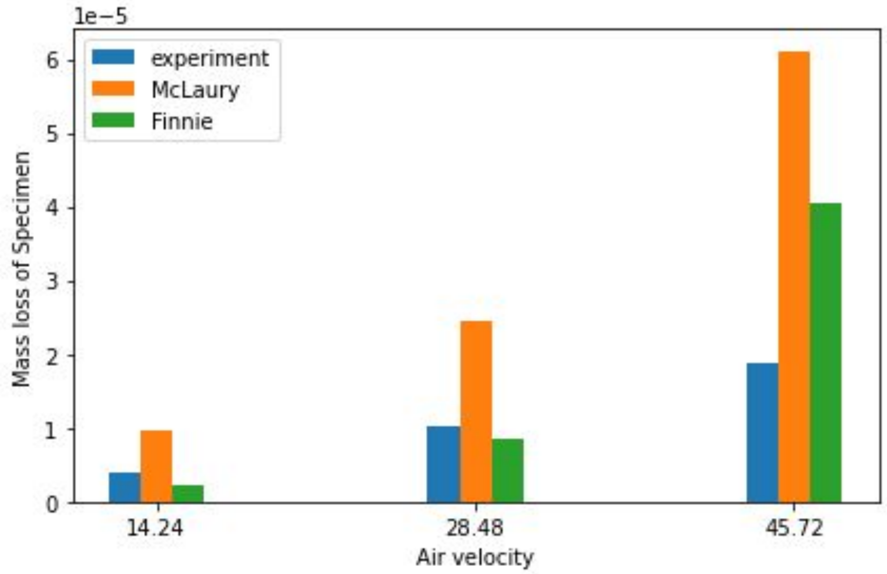


- Optimized mesh number = 500,000

Erosion model Optimization



15.24 m/s 24.48 m/s 45.72 m/s



- Finnie model underestimates ER
- McLaury model overestimates ER
- Finnie model is more accurate, especially at high flow velocity

X. Chen, B.S. Mclaury, S.A. Shirazi, Application and experimental validation of a computational fluid dynamics (CFD)-based erosion prediction model in elbows and plugged tees, Comput. Fluids 33 (10) (2004) 1251–1272.

Automation of the variable setup - running simulation – data saving

```
;boundary conditions inlet setting
/define/boundary-conditions/velocity-inlet inlet n n y y mixture n 70000 n n y 5 10 n

/define/boundary-conditions/velocity-inlet inlet phase-1 n n y y n 12.45

/define/boundary-conditions/velocity-inlet inlet phase-2 n n y y n 12.45 n 0

;boundary conditions inlet_1 setting
/define/boundary-conditions/velocity-inlet inlet_1 mixture n 200000 n n y 5 10 n

/define/boundary-conditions/velocity-inlet inlet_1 phase-1 n n y y n 45

/define/boundary-conditions/velocity-inlet inlet_1 phase-2 n n y y n 45 n 0

;boundary conditions inlet_2 setting
/define/boundary-conditions/velocity-inlet inlet_2 mixture n 200000 n n y 5 10 n

/define/boundary-conditions/velocity-inlet inlet_2 phase-1 n n y y n 45

/define/boundary-conditions/velocity-inlet inlet_2 phase-2 n n y y n 45 n 0

;boundary conditions inlet_3 setting
/define/boundary-conditions/velocity-inlet inlet_3 mixture n 200000 n n y 5 10 n

/define/boundary-conditions/velocity-inlet inlet_3 phase-1 n n y y n 45

/define/boundary-conditions/velocity-inlet inlet_3 phase-2 n n y y n 45 n 0


;# of iteration setup
solve/set/number-of-iterations/200

;initialization
solve/initialize/initialize-flow

;initiate calculation
solve/iterate 200

;dummy
/display/particle-tracks/mixture dpm-erosion-rate-finnie injection-0 () () ()


;overwrite data
file/write-case-data FFF.cas y


;particle velocity data processing
file/export/particle-history-data ensight particle_v particle_v_ensight 10 injection-0 () particle-x-velocity particle-y-velocity particle-z-velocity q 0 1000

;particle position data processing
file/export/particle-history-data ensight particle_p particle_p_ensight 10 injection-0 () particle-x-position particle-y-position particle-z-position q 0 1000
```

FFF.cas
FFF.dat
FFF.jou
FFF.msh

cluster.hpcc.ucr.edu - PuTTY

```
reversed flow in 42 faces on pressure-outlet 10.
  179  1.0022e-04  6.9668e-04  7.0911e-04  5.8706e-04  2.1015e-03  5.2595e-04  0.0000e+00  0:00:01   21

reversed flow in 42 faces on pressure-outlet 10.
  180  9.5011e-05  6.6312e-04  6.7033e-04  5.5956e-04  1.9657e-03  4.8748e-04  0.0000e+00  0:00:01   20

reversed flow in 42 faces on pressure-outlet 10.
  181  9.0266e-05  6.3123e-04  6.3427e-04  5.3262e-04  1.8807e-03  4.5923e-04  0.0000e+00  0:00:01   19

reversed flow in 41 faces on pressure-outlet 10.
  182  8.6112e-05  6.0174e-04  6.0212e-04  5.0708e-04  1.7906e-03  4.3762e-04  0.0000e+00  0:00:01   18

reversed flow in 41 faces on pressure-outlet 10.
  183  8.2091e-05  5.7569e-04  5.7505e-04  4.8345e-04  1.7163e-03  4.1672e-04  0.0000e+00  0:00:01   17

reversed flow in 40 faces on pressure-outlet 10.
  184  7.8538e-05  5.5166e-04  5.5078e-04  4.6093e-04  1.6428e-03  4.0233e-04  0.0000e+00  0:00:01   16

reversed flow in 40 faces on pressure-outlet 10.
  185  7.5217e-05  5.2870e-04  5.2809e-04  4.3919e-04  1.5746e-03  3.8162e-04  0.0000e+00  0:00:01   15
```

Calculation is running in the HPCC in UCR

particle_p.mpg0001	9 KB	1/7/2021 3:51:34 PM
particle_p.mpg0002	9 KB	1/7/2021 3:51:34 PM
particle_p.mpg0003	9 KB	1/7/2021 3:51:34 PM
particle_p.mpg0004	9 KB	1/7/2021 3:51:34 PM
particle_p.mpg0005	9 KB	1/7/2021 3:51:34 PM
particle_p.mpg0006	9 KB	1/7/2021 3:51:34 PM
particle_p.mpg0007	9 KB	1/7/2021 3:51:34 PM
particle_p.mpg0008	9 KB	1/7/2021 3:51:34 PM
particle_p.mpg0009	9 KB	1/7/2021 3:51:34 PM
particle_p.mpg0010	9 KB	1/7/2021 3:51:34 PM
particle_p_ensight.encas	1 KB	1/7/2021 3:51:33 PM
particle_p_ensight.geo	1,188 KB	1/7/2021 3:51:33 PM
particle_p_ensight.new.encas	1 KB	1/7/2021 3:51:34 PM
particle_p_ensight.vel	194 KB	1/7/2021 3:51:33 PM
particle_p_ensight.xml	1 KB	1/7/2021 3:51:33 PM
particle_v.mpg0001	9 KB	1/7/2021 3:51:32 PM
particle_v.mpg0002	9 KB	1/7/2021 3:51:32 PM
particle_v.mpg0003	9 KB	1/7/2021 3:51:32 PM
particle_v.mpg0004	9 KB	1/7/2021 3:51:32 PM
particle_v.mpg0005	9 KB	1/7/2021 3:51:32 PM

Data saved in the directory folder automatically

100 Simulation set with Ansys fluent automation

```
cell_0.0020596
cell_0.0020992
cell_0.0021388
cell_0.0021784
cell_0.0022180
cell_0.0022576
cell_0.0022972
cell_0.0023368
cell_0.0023764
cell_0.0024160
cell_0.0024556
cell_0.0024952
cell_0.0025348
cell_0.0025744
cell_0.0026140
cell_0.0026536
cell_0.0026932
cell_0.0027328
cell_0.0027724
cell_0.0028120
cell_0.0028516
cell_0.0028912
cell_0.0029308
cell_0.0029704
cell_0.0030100
cell_0.0030496
cell_0.0030892
cell_0.0031288
cell_0.0031684
cell_0.0032080
cell_0.0032476
cell_0.0032872
cell_0.0033268
cell_0.0033664
cell_0.0034060
cell_0.0034456
cell_0.0034852
cell_0.0035248
```

```
File is written from fluent in ensight measure
particle coordinates
     180
     901-6.12293e-02-1.00000e-01-1.48773e-03
     902-6.18208e-02-1.00000e-01-3.38328e-03
     903-6.12308e-02-1.00000e-01 1.95927e-03
     904-6.18722e-02-1.00000e-01 3.92511e-03
     905-6.38454e-02-1.00000e-01-6.60504e-03
     906-6.38062e-02-1.00000e-01 6.33036e-03
     907-6.54910e-02-1.00000e-01-7.85739e-03
     908-6.21572e-02-1.00000e-01 3.89291e-04
     909-6.62694e-02-1.00000e-01 8.26049e-03
     910-6.35608e-02-1.00000e-01-3.45795e-03
     911-6.34888e-02-1.00000e-01 4.55221e-03
     912-6.44094e-02-1.00000e-01-4.97850e-03
     913-7.79330e-02-1.00000e-01 3.93421e-03
     914-7.89824e-02-1.00000e-01-1.15054e-03
     915-7.90092e-02-1.00000e-01 9.17172e-04
```

```
Particle Y Velocity
 3.88938e+00 3.65369e+00 3.82533e+00 3.77559e+00 3.84825e+00 3.80954e+00
 3.56383e+00 3.74986e+00 1.21849e+00 4.24176e+00 3.72122e+00 3.68600e+00
 1.99798e+00 1.50128e+00 2.34320e+00 3.54379e+00 9.99426e-01 2.87760e+00
 3.29703e+00 3.88807e+00 3.16737e+00 2.70183e+00 1.18851e+00-7.65068e+00
-2.19491e-01 1.62927e+00 1.36507e+00 3.78422e+00 3.22238e+00 3.91133e+00
 4.15831e+00 3.33980e+00 3.19332e+00 3.72642e+00 3.02259e+00 3.45870e+00
 3.56609e+00 2.07541e+00 3.92664e+00 3.66430e+00 2.93801e+00 3.81474e+00
 4.23211e+00 4.25418e+00 2.99585e+00 3.86736e+00 3.73796e+00 3.78872e+00
 3.56346e+00 2.61576e+00 2.83826e+00 1.14953e+00 4.35200e+00 3.05771e+00
 3.73094e+00 3.94783e+00 2.35619e+00 3.94924e+00 2.31635e+00 3.90841e+00
-2.59208e+00-2.36896e+00-1.53191e+00-2.52393e+00-2.35155e+00-2.50234e+00
-4.27092e+00-1.72286e+00-5.05409e+00-2.40311e+00-2.09608e+00-2.06951e+00
-3.44084e+00-4.35259e+00-2.27154e+00-4.68471e+00-3.32353e+00-5.81406e+00
-3.64474e+00-4.26953e+00-1.96655e+00-3.02828e+00-5.89928e+00-4.21343e+00
-4.79899e+00-4.13226e+00-2.96922e+00-3.92726e+00-4.79330e+00-6.30068e-01
```

```
nodenumber,    x-coordinate,    y-coordinate,    z-coordinate,absolute-pressure,dynamic-pressure,dpm-erosion-rate-finnie
         1,-8.052218691E-02,-2.186964503E-04,-3.999940214E-02, 1.017501686E+05, 1.027968637E+02, 0.000000000E+00
         2,-9.455940122E-02,-2.107107191E-04,-3.999944501E-02, 1.017593760E+05, 9.429612541E+01, 0.000000000E+00
         3,-1.092936108E-02, 2.124682705E-04,-3.999943571E-02, 1.016335186E+05, 2.142853585E+02, 0.000000000E+00
         4, 4.952181219E-02, 2.475428689E-04,-3.999923402E-02, 1.015224712E+05, 3.275329315E+02, 0.000000000E+00
         5, 4.468725752E-02,-3.577437127E-04,-3.999840021E-02, 1.015331995E+05, 3.169053743E+02, 0.000000000E+00
         6, 7.200117131E-02,-4.079770704E-04,-3.999791938E-02, 1.014696444E+05, 3.815776081E+02, 0.000000000E+00
         7, 9.560823398E-02,-5.459903875E-04,-3.999627351E-02, 1.013682931E+05, 4.766563470E+02, 0.000000000E+00
         8, 3.092045335E-02, 5.533126370E-04,-3.999617288E-02, 1.015590559E+05, 2.928202901E+02, 0.000000000E+00
         9, 7.686340264E-02, 5.944170516E-04,-3.999558311E-02, 1.014544219E+05, 4.024516806E+02, 0.000000000E+00
        10,-6.762823209E-02,-6.447709153E-04,-3.999480304E-02, 1.017312173E+05, 1.215971436E+02, 0.000000000E+00
        11,-3.675423947E-02, 6.403410465E-04,-3.999487421E-02, 1.016623187E+05, 1.831331244E+02, 0.000000000E+00
        12, 1.776295780E-02,-6.950465000E-04,-3.999396092E-02, 1.015766512E+05, 2.674614998E+02, 0.000000000E+00
        13,-2.355034135E-02, 6.972941900E-04,-3.999392180E-02, 1.016444009E+05, 2.038637060E+02, 0.000000000E+00
        14,-4.154955688E-02,-7.367415726E-04,-3.999321457E-02, 1.016674613E+05, 1.810248283E+02, 0.000000000E+00
        15,-5.437386840E-02,-8.080094362E-04,-3.999183818E-02, 1.017037808E+05, 1.529970523E+02, 0.000000000E+00
        16, 1.167826911E-02,-8.358463902E-04,-3.999126606E-02, 1.015929703E+05, 2.464960204E+02, 0.000000000E+00
```

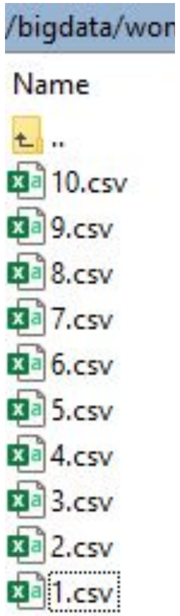
```
/bigdata/wor
Name
..
10.csv
9.csv
8.csv
7.csv
6.csv
5.csv
4.csv
3.csv
2.csv
1.csv
```

100 simulation set varying particle size

Erosion rate, particle position and velocity data of 180 particles (x, y, z, u, v, w)

The data is saved in these csv files with different 10 time steps

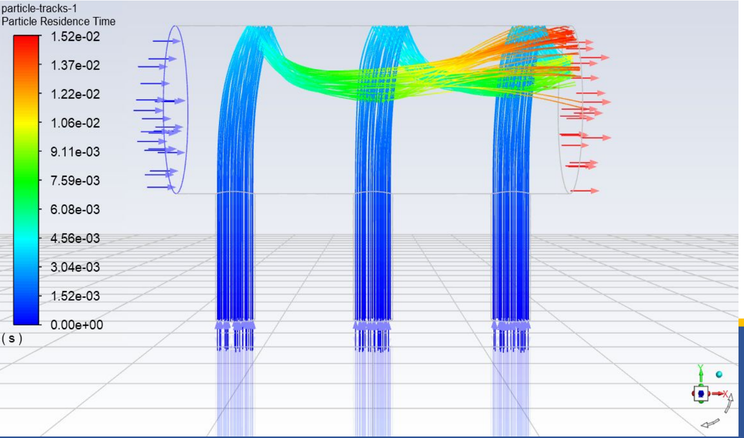
Univariate CNN model for particle position X prediction



	particle	pos_x	pos_y	...	vel_y	vel_z	time
900	901	-0.061229	-0.100000	...	45.00000	0.000000	1
0	901	-0.061207	-0.033156	...	45.21200	-0.191907	2
1080	901	-0.054202	0.027245	...	31.29820	0.158833	3
180	901	-0.034603	0.020200	...	-13.38570	0.449223	4
1260	901	-0.004692	0.010090	...	-1.70672	-0.376443	5
360	901	0.031438	0.010926	...	1.71514	-2.143380	6
540	901	0.069629	0.013032	...	1.13121	-3.476570	7
1620	901	0.100000	0.016345	...	3.88938	-3.750080	8
720	901	0.100000	0.016345	...	3.88938	-3.750080	9
1440	901	0.100000	0.016345	...	3.88938	-3.750080	10

[10 rows x 8 columns]

The saved data in these csv files with different 10 time steps



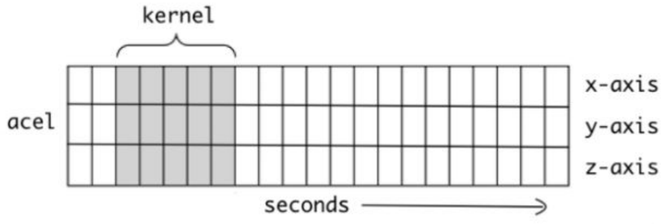
Total 180 particles (particle 901 to particle 1081)
Constructing simple CNN model for particle position prediction, input array is [1st ….. 9th] of x_position and the output is [10th] x_position

Univariate CNN model summary

```
In [5]: model.summary()
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape
=================================================================
conv1d (Conv1D)              (None, 8, 64)
_________________________________________________________________
max_pooling1d (MaxPooling1D) (None, 4, 64)
_________________________________________________________________
flatten (Flatten)            (None, 256)
_________________________________________________________________
dense (Dense)                (None, 50)
_________________________________________________________________
dense_1 (Dense)              (None, 2)
=================================================================
Total params: 13,272
Trainable params: 13,272
Non-trainable params: 0
_________________________________________________________________
```

Sequential: A Sequential model is appropriate for a plain stack of layers where each layer has exactly one input tensor and one output tensor.

Conv1D: Convolution layer is to extract features from an input data.



Maxpooling1D: reducing the spatial size of the convoluted feature with the maximum value in the pool.

Flatten: Flatten the input.

Dense: Dense layer is the regular neural network layer.

Result of particle position prediction with the univariate CNN model

```
[[ 0.01739872]
 [-0.09940347]
 [-0.0314741 ]
 ...
 [ 0.03044665]
 [ 0.02873144]
 [ 0.02846437]]
[ 0.0163451 -0.1       -0.0328619 ...  0.0291407  0.0291407  0.0291407]
Pos_y_Accuracy: 0.99631
[[ 0.01806864]
 [-0.100414  ]
 [ 0.02494469]
 ...
 [ 0.02841071]
 [ 0.02900243]
 [ 0.02924093]]
[ 0.0163451 -0.1        0.027218  ...  0.0291407  0.0291407  0.0291407]
Pos_y_Accuracy: 0.99722
[[ 0.01462822]
 [-0.10090274]
 [-0.03229076]
 ...
 [ 0.02702592]
 [ 0.02816165]
 [ 0.02694769]]
[ 0.0163451 -0.1       -0.0328619 ...  0.0291407  0.0291407  0.0291407]
Pos_y_Accuracy: 0.99635
[[ 0.01815249]
 [-0.09904467]
 [-0.03236211]
 ...
 [ 0.03193789]
 [ 0.0301925 ]
 [ 0.03090305]]
[ 0.0163451 -0.1       -0.0328619 ...  0.0291407  0.0291407  0.0291407]
Pos_y_Accuracy: 0.99634
[[ 0.01984875]
 [-0.09994246]
 [-0.03199193]
 ...
 [ 0.0306093 ]
 [ 0.03111908]
 [ 0.03224027]]
[ 0.0163451 -0.1       -0.0328619 ...  0.0291407  0.0291407  0.0291407]
```

Number of K-fold splits = 10
Activation function = softmax
Epochs = 1000

$$R^2 = \frac{\text{Variance explained by the model}}{\text{Total variance}}$$

R-squared is always between 0 and 100%:

- 0% represents a model that does not explain any of the variation in the response variable around its mean. The mean of the dependent variable predicts the dependent variable as well as the regression model.
- 100% represents a model that explains all of the variation in the response variable around its mean.

	Average r2_score	Stdev of r2_score
X	0.99	0.0031
Y	0.99	0.0007
Z	0.94	0.0008

UCR

Ref: statisticsbyjim

Multivariate CNN model for particle position and velocity prediction

	particle	pos_x	pos_y	...	vel_y	vel_z	time
900	901	-0.061229	-0.100000	...	45.00000	0.000000	1
0	901	-0.061207	-0.033156	...	45.21200	-0.191907	2
1080	901	-0.054202	0.027245	...	31.29820	0.158833	3
180	901	-0.034603	0.020200	...	-13.38570	0.449223	4
1260	901	-0.004692	0.010090	...	-1.70672	-0.376443	5
360	901	0.031438	0.010926	...	1.71514	-2.143380	6
540	901	0.069629	0.013032	...	1.13121	-3.476570	7
1620	901	0.100000	0.016345	...	3.88938	-3.750080	8
720	901	0.100000	0.016345	...	3.88938	-3.750080	9
1440	901	0.100000	0.016345	...	3.88938	-3.750080	10

[10 rows x 8 columns]

Constructing multivariate CNN model for particle position and velocity prediction, input array is [1st ….. 9th] of (x, y, z, u, v, w) and the output is [10th] (x, y, z, u, v, w).

Result of the multivariate CNN model

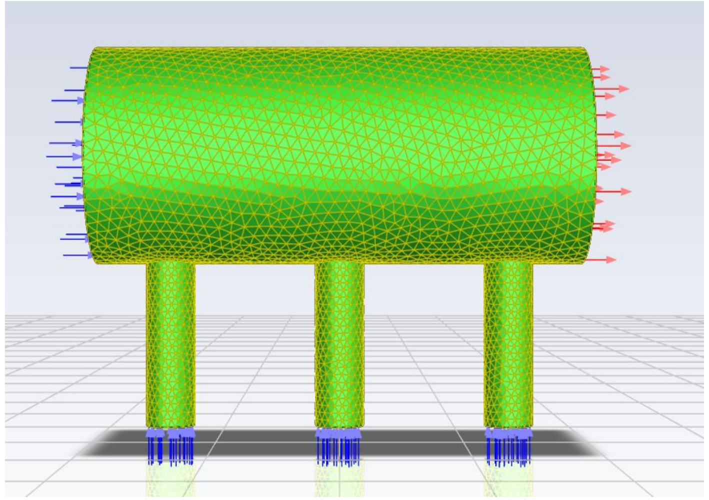
```
[[ 3.5292171e-02  4.7557869e+00]
 [-6.6684738e-02  4.4976635e+01]
 [-6.6684045e-02  4.4976044e+01]
 ...
 [ 2.2473728e-02 -1.5131977e+01]
 [ 2.4155896e-02 -1.5352264e+01]
 [ 2.6693251e-02 -1.5366278e+01]]
[[ 1.63451e-02  3.88938e+00]
 [-1.00000e-01  4.50000e+01]
 [-3.28619e-02  4.56520e+01]
 ...
 [ 2.91407e-02 -1.48001e+01]
 [ 2.91407e-02 -1.48001e+01]
 [ 2.91407e-02 -1.48001e+01]]
r2_score: 0.90515
[[ 3.3687595e-02  4.4856453e+00]
 [-7.0148535e-02  4.4983868e+01]
 [ 2.8719781e-03  3.0673626e+01]
 ...
 [ 2.6130918e-02 -1.4998217e+01]
 [ 3.0141857e-02 -1.4649471e+01]
 [ 2.5309041e-02 -1.4747604e+01]]
[[ 1.63451e-02  3.88938e+00]
 [-1.00000e-01  4.50000e+01]
 [ 2.72180e-02  3.06647e+01]
 ...
 [ 2.91407e-02 -1.48001e+01]
 [ 2.91407e-02 -1.48001e+01]
 [ 2.91407e-02 -1.48001e+01]]
r2_score: 0.90162
```

Number of K-fold splits = 10
Activation function = softmax
Epochs = 1000

	Average r2_score	Stdev of r2_score
x/u	0.90	0.05
y/v	0.91	0.01
z/w	0.71	0.01
xyz/uvw	0.84	0.03

Conclusion & Future Plans

Erosion location and erosion rate prediction based on the particle tracking data



The number of meshes (cells) on the surface geometry: 8012

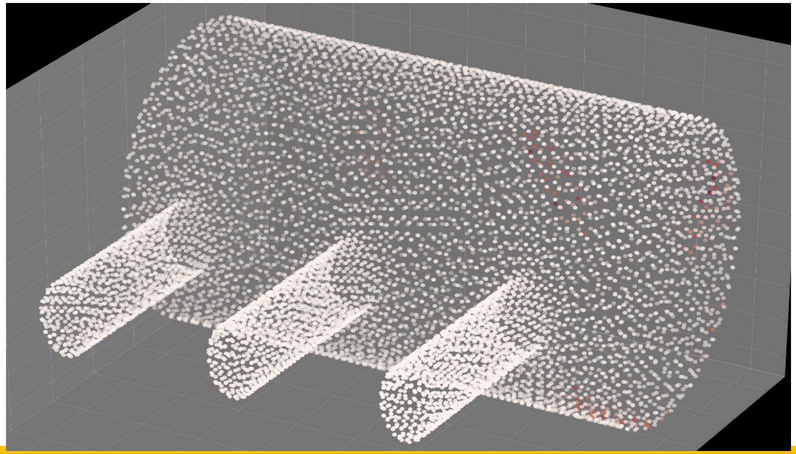
Each cell has its own cartesian coordinates and the erosion rate after the calculation from Ansys fluent.

Input shape

[x, y, z, u, v, w]	time steps	Particle #	# of simulations
6	50	180	1000

CNN

Erosion prediction



Normalization of the erosion value data

Part of the erosion data sample

```
7827,-7.229450624E-03,-8.475839347E-02, 6.794056389E-03, 0.000000000E+00
7828, 8.885966986E-02,-1.185526699E-02,-3.814418986E-02, 0.000000000E+00
7829,-5.933347275E-04,-3.646857291E-02, 1.634321176E-02, 0.000000000E+00
7830,-7.514000684E-02, 3.989124298E-02,-2.231882419E-03, 0.000000000E+00
7831,-6.330247968E-02,-5.302060023E-02, 7.324883249E-03, 0.000000000E+00
7832,-7.987599075E-02,-6.079739332E-02, 1.147684641E-03, 0.000000000E+00
   1, 1.000000015E-01,-2.099469304E-02,-2.354580723E-02, 0.000000000E+00
   2, 1.000000015E-01,-2.381087840E-02,-2.243536897E-02, 0.000000000E+00
   3, 1.000000015E-01, 2.230577730E-02, 3.106298298E-02, 0.000000000E+00
   4, 1.000000015E-01, 2.338075824E-02, 2.847195975E-02, 0.000000000E+00
   5, 1.000000015E-01, 2.296154387E-02,-3.123693727E-02, 0.000000000E+00
   6, 1.000000015E-01, 1.929423586E-02,-2.853380330E-02, 0.000000000E+00
   7, 1.000000015E-01, 2.513688616E-02,-2.237791382E-02, 0.000000000E+00
   8, 1.000000015E-01, 2.212263085E-02,-2.423404902E-02, 0.000000000E+00
   9, 1.000000015E-01, 3.463008616E-04,-3.806517646E-02, 5.349202450E-05
```

Normalized erosion rate

```
269 0.04826442881379604
302 0.1165303875830884
317 0.030835448383175097
318 0.03968942049945025
331 0.05954603632759854
533 0.05822082226337687
534 0.062324094144239464
781 0.11481442982896198
812 0.05342556562353167
813 0.09878924784196
901 0.09385222362249422
948 0.05153115135671478
959 0.14195910388411778
1001 0.13425618063500655
1002 0.06836045033313315
1035 0.06423311836701352
1037 0.039939926670106894
1051 0.06797445930351068
1058 0.06568821006698913
1061 0.06670443497889313
```

Data dimension: 2D
X: (x, y, z, Erosion rate)
Y: Number of cells

Cells with non-zero values: 147
Cells with zero: 7,865

Maximum value of the erosion rate: 4.74 x 10^-4 kg/m^2 · s

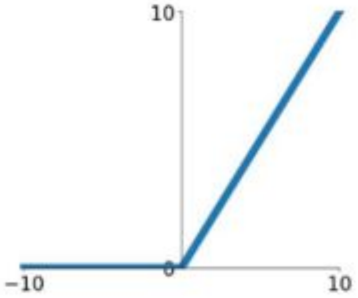
Erosion rate is too small for the input of CNN → Normalization

$$x_{\text{norm}} = \frac{x - \min(x)}{\max(x) - \min(x)}$$

Activation function of the CNN model for erosion prediction

ReLU

$\max(0, x)$



Rectified Linear unit, Computationally efficient
Non-linear (It has derivative function and allows Backpropagation)

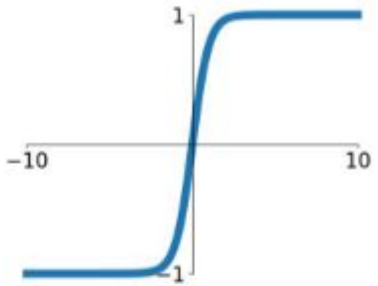
But the network cannot perform backpropagation when the inputs approach zero. (Dying ReLU problem)

Softmax $\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$

Softmax turns arbitrary real values into probabilities. The outputs of the Softmax transform are always in the range [0, 1] and add up to 1. Hence, they form a probability distribution. (not good for erosion prediction)

tanh

$\tanh(x)$



non-linear activation function
Zero centered
Could have vanishing gradient problem

CNN model summary for erosion prediction

Model: "sequential_1"

Layer (type)	Output Shape	Param #
conv3d (Conv3D)	(None, 50, 180, 6, 10)	280
max_pooling3d (MaxPooling3D)	(None, 25, 90, 3, 10)	0
dropout (Dropout)	(None, 25, 90, 3, 10)	0
conv3d_1 (Conv3D)	(None, 25, 90, 3, 20)	1620
max_pooling3d_1 (MaxPooling3	(None, 13, 45, 2, 20)	0
dropout_1 (Dropout)	(None, 13, 45, 2, 20)	0
conv3d_2 (Conv3D)	(None, 13, 45, 2, 40)	6440
max_pooling3d_2 (MaxPooling3	(None, 6, 22, 1, 40)	0
flatten (Flatten)	(None, 5280)	0
dense (Dense)	(None, 8012)	42311372

- 3 Consecutive convolution3D & Maxpooling3D with padding with increasing number of filters
- Activation function used: tanh
- Flatten into dense layer with 8012 in order to get the dimension of the erosion input data

Epochs: 100
Number of K-fold splits = 5
CNN model evaluation method = r2_score

Summary

Hybrid approach of combining CFD and machine learning models to build a predictive modeling pipeline for erosion rates.

CFD Modeling

* Modeling fluid flow
*Modeling particle transport
*Input data:
- -CAD model for elbow
- -Initial flow rates
- -Boundary conditions

*Output :
- -**Particle trajectory**
- -Erosion rates

→

Particle trajectory prediction

CNN model
* Input :
- -**Particle trajectories**

*Output:
- CNN model for particle trajectory prediction

→

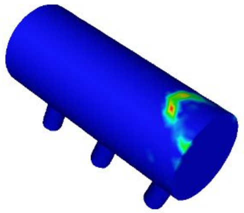
Erosion rate prediction

CNN model
* Input :
- -Initial conditions
- -Particle properties
- -**Particle trajectories**
- -**Erosion rate**

*Output:
- CNN model for erosion rate predictions

Summary of current progress

<u>CFD Modeling</u>



* Modeling fluid flow

*Modeling particle transport

- Optimized mesh density
- Optimized erosion model → **<u>Finnie</u>**
- Preliminary results
    - Connection angle vs. Sum erosion rate

      → **Optimal connection angle = 40°**
    - Pipe diameter vs. Sum erosion rate

      → **Quadratic proportional**

Done

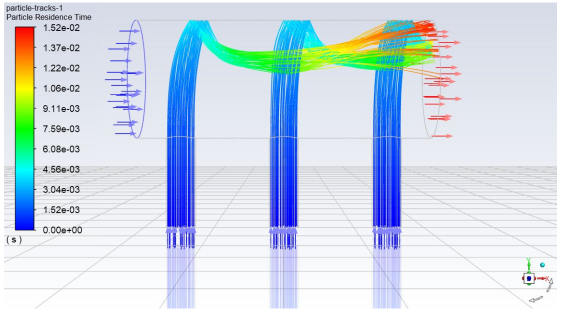
<u>Particle trajectory prediction</u>

* CNN model

*Input : Initial particle velocities/positions

*Output : particle velocities/positions

***R2 score = 0.84**



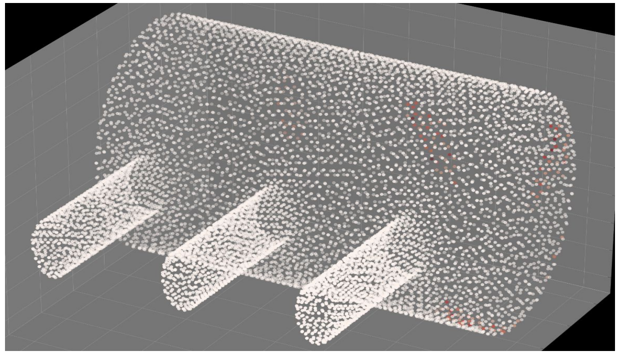
Done

<u>Erosion rate prediction</u>

* CNN model

*Input : Particle trajectories

*Output : Erosion rate



In progress

Conclusion

(1) CFD calculations to simulate fluid flow and particle transport in a boiler header.

(2) CFD results are considered as the input parameter

(3) A new model based on statistical analysis of CFD output parameters is developed.

- CNN models are explored to act as a black box

- CFD output is taken as input parameter and erosion is the output.

- A multilayer feed-forward network with a back-propagation algorithm is implemented

- CNN has potential to be developed further as a tool to predict erosion rate.

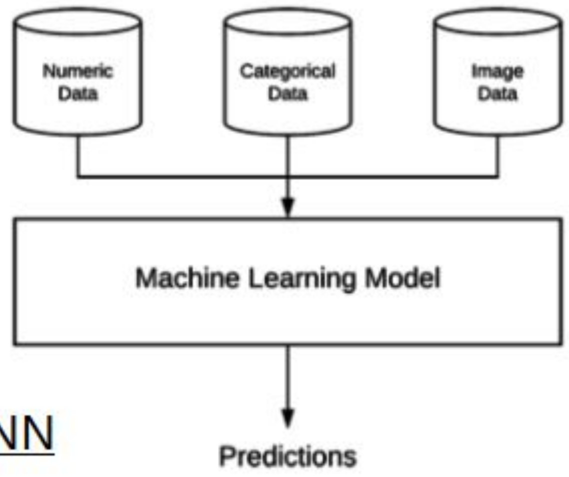
Future plan

- Any statistical model, more data points lead to better models

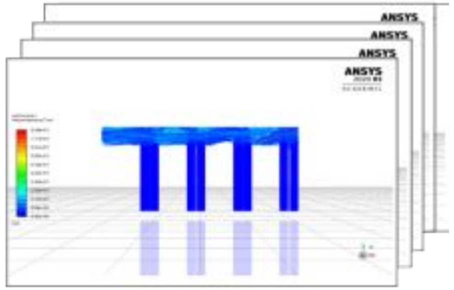
- This can be achieved by more CFD calculations

- Correlation analysis will be performed on CFD output parameters to identify ones affecting erosion rate prediction the most.

Particle size
Mass flow rate
Flow velocity
Surface shear stress
...



Multi-input CNN



Time series particle tracking data

Thank you!

Supported by UCR/HBCU DE-FE0031746

Disclaimer

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.