

# Predicting Next Steps of a CFD Simulation using Deep Learning

Janaina Gomide<sup>1</sup>, Raquel Lobosco<sup>1</sup>, Guilherme Antônio Santos<sup>1</sup>

<sup>1</sup>Instituto Politécnico - Universidade Federal do Rio de Janeiro (UFRJ)

{janainagomide, raquellobosco, burgaoguilherme}@gmail.com

**Abstract.** *Computational fluid dynamics (CFD) simulations have applications in several areas such as the aeronautical, automotive and energy industries. Fluid flows are described by equations whose resolution is still limited by computational cost. The goal of this work is to complement CFD simulations in terms of predicting subsequent instants of the simulation using deep learning. The proposed methodology models this problem as predicting the next frame of a video. The input to the deep network is the first images of the CFD simulation and the output is the next images. The results obtained for simulating a flow through of a fixed cylinder showed SSIM values greater than 0.83.*

**Resumo.** *As simulações de dinâmica dos fluidos computacional (CFD) possuem aplicações em diversas áreas como indústria aeronáutica, automobilística e energética. Os fluxos de fluido são descritos por equações cuja resolução ainda é limitada pelo custo computacional. O objetivo desse trabalho é complementar as simulações de CFD em termos de previsão dos instantes subsequentes da simulação utilizando aprendizado profundo. A metodologia proposta modela esse problema como a previsão do próximo quadro de um vídeo. A entrada para a rede profunda são as primeiras imagens da simulação CFD e a saída são as próximas imagens. Os resultados obtidos para a simulação de um fluxo através de um cilindro fixo apresentaram valores de SSIM maiores que 0.83.*

## 1. Introdução

As simulações de Dinâmica dos Fluidos Computacional (CFD) desempenham um papel essencial na modelagem de fenômenos físicos complexos porque contribuem para o desenvolvimento tecnológico de áreas estratégicas da Engenharia, tal como para a indústria aeronáutica e automobilística. Os fluxos de fluido são descritos pelas equações de conservação da massa, da quantidade de movimento e da energia e a resolução dessas equações diferenciais pode ser de alto custo computacional [Yeung et al. 2015].

No cotidiano, a solução das equações de Navier-Stokes em todas as escalas ainda permanece uma alternativa inviável computacionalmente e os modelos de turbulência mais utilizados ainda são os modelos de média de Reynolds [Vinuesa and Brunton 2022a]. Neste contexto, o recente desenvolvimento dos métodos de aprendizado de máquinas motivou a combinação de técnicas estatísticas com métodos baseados em modelos e permitiu uma nova perspectiva para a modelagem de sistemas dinâmicos complexos. Criou-se assim, uma alternativa capaz de contribuir para a redução do custo numérico da resolução das equações da fluidodinâmica [Ozaki and Aoyagi 2022].

Esses dois domínios da ciência, que pareciam separados até muito recentemente, permitem agora integrar alternativas para a representação dos fenômenos físicos [Sirignano et al. 2019]. Enquanto o aprendizado de máquinas enfrenta o desafio de demandar um grande quantitativo de dados, a representação fluidodinâmica gera, através da resolução das equações diferenciais, um quantitativo massivo de imagens instantâneas que representam campos de propriedades físicas variáveis ao longo da posição e do tempo dentro no domínio computacional.

O objetivo desse trabalho é complementar os modelos de fluidodinâmica computacional tradicionais em termos de previsão dos instantes subsequentes da simulação numérica utilizando aprendizado profundo.

Esse trabalho modela a previsão de um passo de tempo de uma simulação CFD no intuito de prever o instante subsequente, ou seja, prever o que acontece no futuro em forma de uma imagem do campo de velocidade. Essa previsão é feita com base em informações de imagens anteriores do que aconteceu até o presente momento. A previsão do próximo quadro refere-se a começar com quadros de vídeos e construir uma rede que pode gerar de forma precisa os quadros subsequentes. A entrada da rede são alguns quadros anteriores e a previsão são os próximos quadros, conforme apresentado em [Zhou et al. 2020].

Na seção seguir são apresentados alguns trabalhos relacionados que utilizam o aprendizado de máquina em simulações CFD, na Seção 3 é descrita a simulação CFD utilizada como estudo de caso nesse trabalho. A Seção 4 apresenta a metodologia proposta e os resultados obtidos são apresentados na Seção 5. A conclusão e possíveis trabalhos futuros são apresentados na Seção 6.

## 2. Trabalhos relacionados

Há diversos trabalhos que aplicam o aprendizado de máquina no contexto de simulações CFD. Nessa seção alguns trabalhos são destacados e relacionados ao trabalho proposto nesse artigo.

O trabalho apresentado em [Vinuesa and Brunton 2022b] destaca algumas áreas em que o aprendizado de máquina pode ser aplicado a simulações CFD como por exemplo em simulações utilizando média de Reynolds, para melhorar modelos de turbulência em simulações de grandes escalas e inclusive para acelerar simulações numéricas diretas.

Uma possível abordagem utiliza o aprendizado de máquina informado pela física ou *Physics-Informed Machine Learning* (PIML) que são ferramentas de aprendizado de máquina que aproximam a solução numérica de equações de modelo, como equações diferenciais parciais, adicionando-as de alguma forma como um componente. Em [Mathias et al. 2022a] os autores implementaram uma Rede Neural Informada por Física (PINN) para resolver as equações bidimensionais de Burgers e em [Mathias et al. 2022b] descrevem uma PINN que simula o fluxo induzido pela maré em um canal de porto.

Em [Kochkov et al. 2021] os autores usam o aprendizado profundo para melhorar as aproximações dentro da dinâmica de fluidos para modelar fluxos turbulentos bidimensionais. São criados modelos tanto para simulação numérica direta de turbulência quanto para simulação de grande vortividade (*large-eddy*). Os resultados são tão precisos quanto aos comparados com *solvers* com resolução até 10 vezes mais fina em cada dimensão espacial, que resultou em melhoria de 40 a 80 vezes de processamentos computacionais.

A previsão de campos de fluxo de simulação CFD de aerofólios foi apresentada em [Bhatnagar et al. 2019]. Nesse trabalho os autores conseguiram estimar efetivamente as ordens de magnitude do campo de velocidade e pressão mais rápido do que a solução das equações diferenciais por média de Reynolds, tornando possível estudar o impacto da forma do aerofólio e das condições operacionais nas forças aerodinâmicas e o campo de fluxo quase em tempo real.

Outra possível aplicação é uma abordagem híbrida, proposta em [Usman et al. 2021], que envolve a inicialização com uma simulação CFD e a solução é prevista por um modelo de aprendizado de máquina para obter uma convergência mais rápida que o CFD tradicional. A simulação analisada é do fluxo através de um cilindro fixo, são utilizadas dez imagens para prever a décima primeira.

Inspirado nestes trabalhos, o presente artigo propõe um modelo de aprendizado profundo para gerar os próximos passos da simulação CFD do fluxo através de um cilindro fixo. A metodologia propõe a modelagem de um problema de previsão do próximo quadro de um vídeo. A rede neural utilizada é uma rede neural recorrente que combina os princípios das redes convolucionais, como explicado nas próximas seções desse artigo.

### 3. O modelo fluidodinâmico

O presente artigo aborda a previsão do fluxo transiente bidimensional através de um cilindro fixo usando um modelo de aprendizado profundo e investiga a precisão do campo de velocidade previsto para o instante subsequente da simulação. O modelo gera os componentes  $x$  e  $y$  do campo de velocidade do fluxo e a precisão é investigada com foco na imagem do perfil de velocidade do escoamento do fluido.

Para descrever o movimento do fluido Newtoniano com viscosidade constante são utilizadas duas equações diferenciais. A Equação de conservação da massa, Equação 1,

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \quad (1)$$

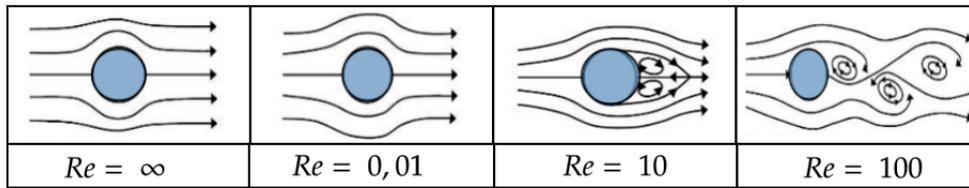
em que  $u$  é a velocidade,  $\rho$  é a massa específica,  $t$  é o tempo. E a denominada Equação geral do transporte, Equação 2,

$$\frac{\partial (\rho \phi)}{\partial t} + \nabla \cdot (\rho \phi \mathbf{u}) = \nabla \cdot (\Gamma \nabla \phi) + S_\phi \quad (2)$$

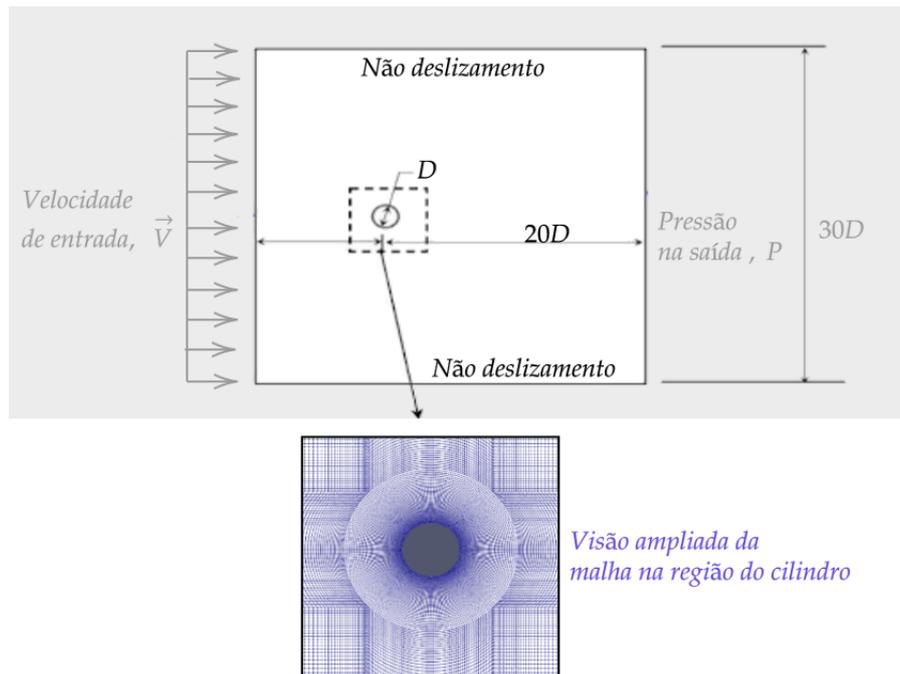
em que  $\Gamma$  é o coeficiente de difusão,  $S_\phi$  é o termo fonte,  $u$  é a velocidade,  $\rho$  é a massa específica,  $t$  é o tempo,  $\phi$  é a propriedade intensiva da quantidade de movimento. A Equação 2 também é conhecida como Equação de Navier Stokes.

O escoamento ao redor de um cilindro é uma abordagem clássica em mecânica dos fluidos, por apresentar característica geométrica corriqueira nas aplicações da engenharia [Usman et al. 2021].

Quando o número de Reynolds aumenta, o fluxo de esteira através do cilindro varia e o desprendimento alternado de vórtices causa pressões flutuantes na superfície sólida [Ye et al. 2020]. A Figura 1 é uma ilustração esquemática do perfil de escoamento sobre um cilindro em função do número de Reynolds.



**Figure 1. Ilustração esquemática do perfil de escoamento sobre um cilindro em função do número de Reynolds.**



**Figure 2. Domínio computacional e ilustração da malha próxima a região do cilindro.**

Para a modelagem fluidodinâmica foi utilizado o solver simpleFoam do Software livre de código aberto OpenFoam<sup>1</sup>. O domínio computacional, as condições de contornos e o refinamento da malha próximo da região da parede, foram definidas conforme demonstra a Figura 2. A condição de velocidade de entrada foi fixada de acordo com o número de Reynolds e a Tabela 3 descreve as condições de contorno para a pressão e para velocidade em cada uma das faces.

Osbourne Reynolds estudou a transição entre os regimes de escoamento laminar e turbulento e definiu o critério que representa a relação entre as forças viscosas e inerciais Equação 3.

$$Re = \frac{\rho V D}{\mu} \quad (3)$$

Foram utilizadas na simulação a viscosidade e a massa específica da água a 20°C em uma variação do número de Reynolds entre 90 e 500 que permitiu a análise e previsão do desprendimento dos vórtices em função das instabilidades hidrodinâmicas.

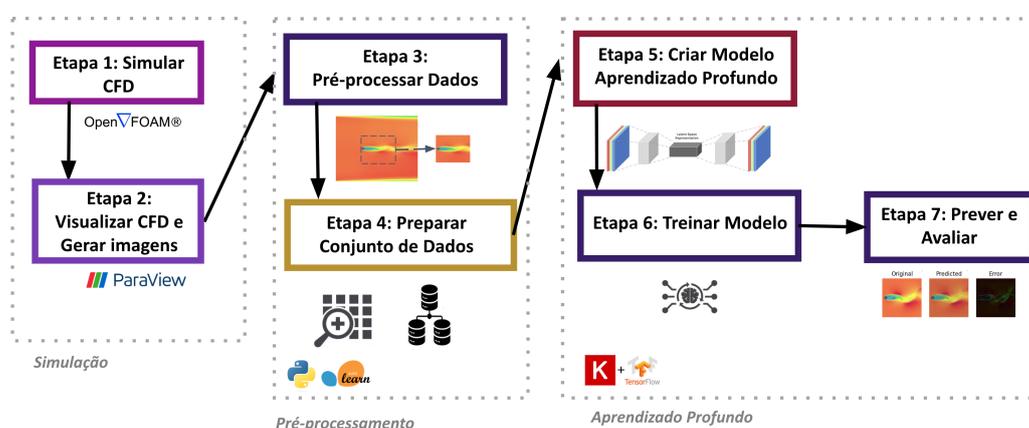
<sup>1</sup>OpenFoam: <https://www.openfoam.com/>

Parâmetro	U	P
inlet	Valor variante com o número de Reynolds	Gradiente zero
outlet	Gradiente zero	Gradiente zero
wall	Nao deslizamento	Gradiente zero
obstacle	Nao deslizamento	Gradiente zero
frontAndBack	Vazio	Vazio

**Table 1. Condições de contorno pra definição das fronteiras do domínio.**

## 4. Metodologia

A metodologia proposta nesse trabalho envolve as etapas descritas na Figura 3. O conjunto de dados utilizado e o código implementado em Python estão disponíveis no GitHub<sup>2</sup>. A seguir são apresentadas em detalhes as etapas.



**Figure 3. Etapas da metodologia proposta para prever os próximos instantes da simulação CFD.**

### 4.1. Simulação CFD e geração de imagens

Para iniciar é necessário definir o solver para a simulação CFD e executá-la (Etapa 1). Nesse trabalho a simulação a ser utilizada é a previsão do fluxo através de um cilindro fixo, conforme descrito na Seção 3. Nessa etapa é utilizado o solver SimpleFoam do software OpenFoam para realizar a simulação CFD.

Após a simulação CFD, é necessário visualizá-la em um mapeamento de imagens (Etapa 2). Nessa etapa o software livre ParaView<sup>3</sup> é utilizado para gerar as imagens da simulação. Esse software de visualização científica é capaz de lidar com uma ampla variedade de tipos de dados, incluindo dados estruturados e não estruturados, dados de malha, dados de partículas e dados em tempo real. Além disso, oferece recursos para visualizar e analisar dados complexos, incluindo dados numéricos, dados de simulação, dados experimentais, entre outros.

Foi gerado um total de 42 simulações CFD, cada simulação foi realizada para um número de Reynolds diferente, que variou de 90 a 500. Para cada simulação foram

<sup>2</sup><https://github.com/mlandengineering/nextStepsCFDSimulationCylinder/>

<sup>3</sup>ParaView: <https://www.paraview.org/>

geradas 50 imagens em RGB no formato 954 x 954 e somente as 20 finais da simulação são utilizadas para entrada para a próxima etapa da metodologia. São utilizadas apenas as 20 finais para garantir que o modelo de fluidodinâmica computacional ficou estável.

## 4.2. Pré-processamento e estruturação do conjunto de dados

A Etapa 3 da metodologia envolve realizar um pré-processamento nas imagens para transformá-las de 954 x 954 para o formato de 128 x 128 que é o formato de entrada da rede neural profunda.

A Figura 4 apresenta seis exemplos do conjunto de dados, cada exemplo é gerado por uma simulação com um número de Reynolds diferente. O conjunto de dados é formado por 42 exemplos (vídeos) e cada exemplo possui 20 quadros. Das 42 simulações, foram selecionadas de forma aleatória 28 para treinar o modelo, 6 para validação e 8 para teste.

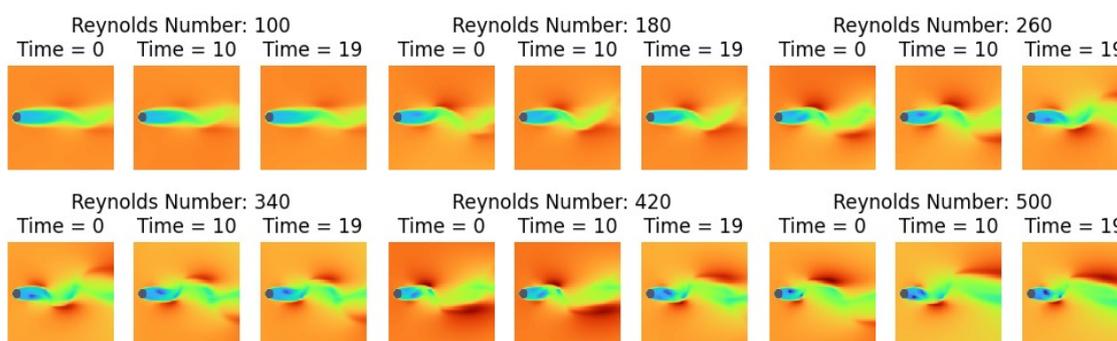


Figure 4. Exemplos do conjunto de dados gerado pela simulação CFD.

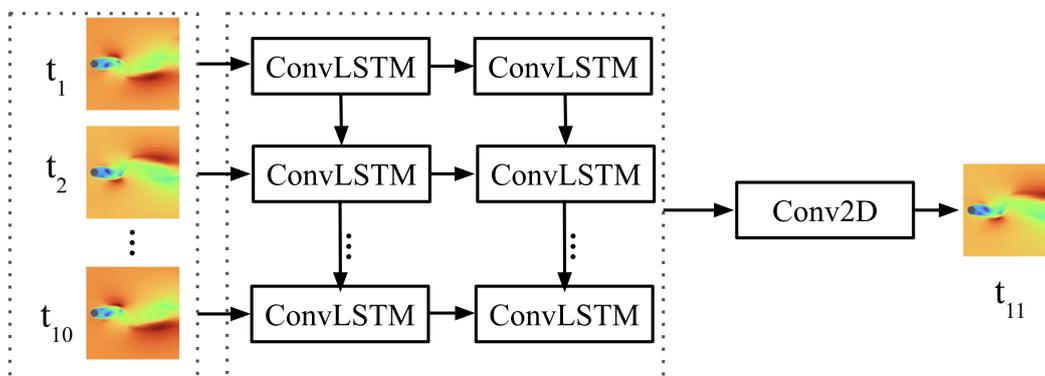
## 4.3. Aprendizado profundo

A arquitetura utilizada para o aprendizado profundo é a Rede de Convolução LSTM (ConvLSTM), ou *Convolutional Long Short-Term Memory*, proposta em [Shi et al. 2015]. É uma arquitetura de rede neural recorrente que combina os princípios das redes convolucionais (CNNs) e das redes LSTM (Long Short-Term Memory), introduzindo uma célula recorrente convolucional em uma camada LSTM. A ConvLSTM é especialmente projetada para processar dados sequenciais e nesse trabalho, essa rede é utilizada para prever o próximo quadro, ou seja, o processo de prever quais quadros do vídeo virão a seguir, dada uma série de quadros anteriores.

A rede ConvLSTM recebe uma sequência de imagens como entrada, representando os quadros anteriores, e emite uma sequência diferente de imagens, representando os quadros previstos. A sequência de quadros é representada por um tensor, com dimensões  $(Q, A, L, C)$  em que  $Q$  é o número de quadros,  $C$  é o número de canais,  $A$  é a altura de cada quadro e  $L$  é a largura de cada quadro. A saída tem todas as mesmas dimensões que a entrada, com exceção de  $T$ , que pode variar. Para o conjunto de dados que será utilizado nesse trabalho os valores são  $Q = 10$ ,  $A = 128$ ,  $L = 128$  e  $C = 3$ .

Para treinar a rede ConvLSTM, foram escolhidos de forma aleatória 10 quadros entre 20 quadros gerados e a rede deve prever o 11º quadro. As configurações do processo de treinamento foram: otimizador Adam, 200 epochs e a função de erro na medida do

índice de similaridade estrutural (*Structural Similarity Index Measure*), pois no conjunto de dados original, cada valor de pixel é 0 até 1, ou seja, é um problema de regressão do valor do pixel. A Figura 5 apresenta a arquitetura da rede.



**Figure 5. Arquitetura da Rede ConvLSTM utilizada para prever o próximo quadro da simulação CFD.**

A saída da rede produz um único quadro. Para gerar os próximos nove quadros do vídeo que compõem os instantes finais da simulação CFD e visualizar o que o modelo aprendeu, será utilizada a seguinte abordagem para testar o modelo. O teste se inicia utilizando os 10 quadros iniciais do exemplo como entrada e para o modelo prevê o 11º quadro; logo em seguida, são fornecidos os quadros de 2 a 11 como entrada e o modelo deve prevê o 12º quadro e assim por diante. Dessa forma são gerados os 10 próximos quadros do exemplo, um a um.

Uma vez que os resultados finais do modelo de aprendizado profundo são imagens, as métricas utilizadas para avaliar a qualidade de imagens e comparar a imagem original que gerada pela simulação com a imagem prevista pela rede são Peak Signal-to-Noise Ratio (PSNR) e Structural Similarity Index (SSIM), [Wang et al. 2004]. O PSNR fornece uma medida quantitativa da qualidade da imagem em termos de erro médio, sendo maior o valor do PSNR, melhor a qualidade da imagem. O PSNR é calculado conforme a Equação 4.

$$PSNR = 10 \cdot \log_{10} \left( \frac{\max^2}{MSE} \right) \quad (4)$$

Em que:  $\max$  representa o valor máximo possível que um pixel pode ter na imagem e  $MSE$  é o Erro Médio Quadrático entre as imagens original e processada.

O SSIM é uma métrica de similaridade estrutural que leva em consideração não apenas o erro de pixel, mas também a estrutura e a percepção visual da imagem. Ele avalia a similaridade entre duas imagens com base na luminância, contraste e estrutura. O SSIM varia de 0 a 1, onde 1 indica uma imagem idêntica à imagem original. A fórmula do SSIM é dada pela Equação 5.

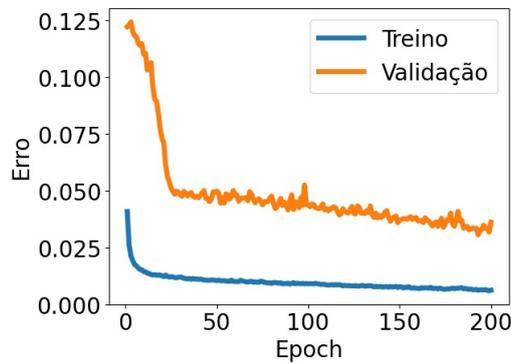
$$SSIM(x, y) = \frac{(2 \cdot \mu_x \cdot \mu_y + C_1) \cdot (2 \cdot \sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1) \cdot (\sigma_x^2 + \sigma_y^2 + C_2)} \quad (5)$$

Em que:  $\mu_x$  e  $\mu_y$  são as médias das imagens  $x$  e  $y$ , respectivamente;  $\sigma_x$  e  $\sigma_y$  são os desvios padrão das imagens  $x$  e  $y$ ;  $\sigma_{xy}$  é a covariância das imagens  $x$  e  $y$  e  $C_1$  e  $C_2$  são constantes.

Além disso, será realizada uma avaliação dos resultados de forma qualitativa, examinando visualmente os quadros gerados para verificar se são uma continuação coerente dos dados de entrada.

## 5. Resultados

A Figura 6 apresenta a função de erro para o treino e a validação durante o treinamento da rede ConvLSTM. A medida de erro utilizada é  $(1 - SSIM)$ . Observe que depois de 25 epochs já há uma melhora significativa na medida de erro.



**Figure 6. Valores de erro no treino e validação durante o treinamento da rede.**

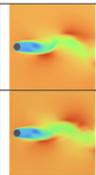
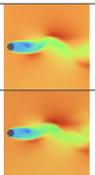
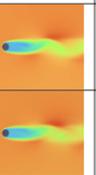
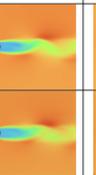
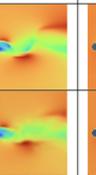
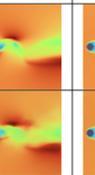
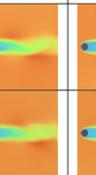
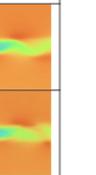
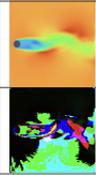
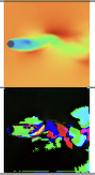
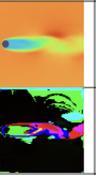
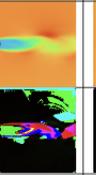
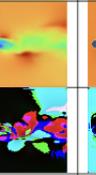
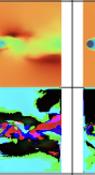
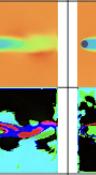
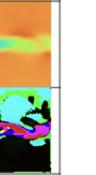
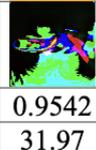
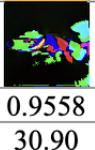
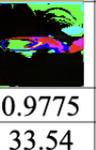
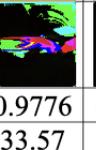
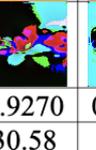
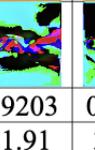
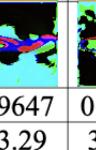
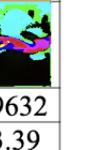
A Tabela 2 apresenta as métricas obtidas pelo modelo ao prever as dez imagens dos oito exemplos de teste. Os valores das métricas SSIM e PSNR para a primeira imagem gerada pela rede, para a última imagem (décima) e a média dos valores de todas as imagens e desvio padrão podem ser visualizados nessa tabela. Observe que os valores de SSIM foram todos superiores a 0.82. Além disso, os valores de SSIM para a primeira e para a última imagem gerada são muito próximos. O SSIM médio das imagens também é alto e o desvio padrão baixo. Para a métrica PSNR os valores ficaram entre 28 e 33 aproximadamente, representando também um excelente resultado.

**Table 2. Valores das métricas SSIM e PSNR para os oito exemplos de teste. Os valores são para a primeira e última imagens e a média e o desvio padrão de todas as dez imagens geradas.**

Imagens	Métricas	Exemplos de teste							
		0	1	2	3	4	5	6	7
Primeira	SSIM	0.9542	0.8312	0.9775	0.8935	0.9270	0.9647	0.8828	0.8677
	PSNR	31.97	29.71	33.54	30.63	30.58	33.29	31.39	29.82
Última	SSIM	0.9558	0.8609	0.9776	0.9045	0.9203	0.9632	0.8686	0.8909
	PSNR	30.90	28.09	33.57	30.02	31.91	33.39	30.34	31.10
Média (desvio) das 10 imagens	SSIM	0.9539	0.8424	0.9764	0.9019	0.9160	0.9634	0.8805	0.8846
	PSNR	(0.0018)	(0.0156)	(0.0010)	(0.0062)	(0.0091)	(0.0018)	(0.0101)	(0.0095)
		31.85	29.22	33.65	29.92	30.39	33.23	30.68	30.70
		(0.7827)	(0.7456)	(0.2536)	(0.7132)	(0.9031)	(0.2549)	(0.5975)	(0.4999)

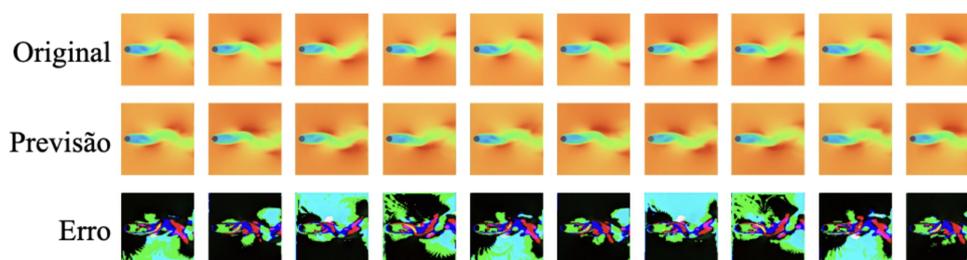
A Figura 7 apresenta a primeira e a última imagem de alguns exemplos de teste. Para cada caso tem a imagem original que foi gerada pela simulação CFD, a imagem prevista pelo algoritmo de aprendizado profundo e a imagem de erro que é a diferença entre

essas duas imagens. Além disso, a Figura 7 apresenta também as métricas de avaliação. As imagens geradas apresentam bastante similaridade visual com as originais e o erro acontece principalmente na região próxima no desprendimento do vórtice.

	Exemplo 0		Exemplo 2		Exemplo 4		Exemplo 5	
	Primeira	Última	Primeira	Última	Primeira	Última	Primeira	Última
Original								
Previsão								
Erro								
SSIM	0.9542	0.9558	0.9775	0.9776	0.9270	0.9203	0.9647	0.9632
PSNR	31.97	30.90	33.54	33.57	30.58	31.91	33.29	33.39

**Figure 7. A primeira e a última imagem de alguns dos exemplos de teste e as métricas de avaliação.**

As dez imagens geradas para todo um exemplo de teste podem ser visualizadas na Figura 8. É possível avaliar que as imagens previstas pelo algoritmo são muito similares às imagens originais e, além disso, são uma continuação da primeira imagem. Esse é o exemplo 0 da Tabela 2. É interessante notar a alternância do erro de acordo com o desprendimento do vórtice.



**Figure 8. Todos os frames do exemplo 0 com as imagens originais da simulação CFD, as imagens previstas pelo algoritmo e a imagem da diferença entre elas.**

## 6. Conclusão e Trabalhos Futuros

Nesse trabalho foi realizada a previsão de instantes subsequentes de uma simulação de CFD. A simulação realizada foi de um fluxo transiente bidimensional através de um cilindro fixo. Foram geradas 42 simulações com diferentes números de Reynolds, o que permitiu avaliar diferenças nas instabilidades hidrodinâmicas.

A previsão dos próximos instantes de uma simulação foi modelada como o problema de prever os próximos quadros de um vídeo e foi utilizada a estrutura da rede neural profunda ConvLSTM para prever a próxima imagem dada uma sequência de 10 imagens como entrada.

Os experimentos foram realizadas com 42 simulações, sendo cada uma composta por 20 quadros. Desses quadros, 10 foram utilizados para entrada da rede para poder

prever os próximos 10 quadros. Os resultados obtidos ao testar o modelo de aprendizado obtiveram valores de SSIM e PSNR próximos para todas as dez imagens geradas e a média dos valores de SSIM para todas as dez imagens foi maior que 0.84 para todos os oito exemplos de teste. O melhor resultado foi de 0.96 para um exemplo de teste. As avaliações visuais das imagens geradas representam uma continuação da simulação e formam uma sequência lógica dos próximos passos.

Foi interessante perceber nos resultados a flutuação do erro de acordo com a direção do desprendimento do vórtice dentro do domínio computacional.

Como trabalhos futuros pretende-se avaliar diferentes redes de aprendizado profundo e outras simulações de fluidodinâmica. Uma possível rede a ser considerada é a FutureGAN, proposta em [Aigner and Körner 2019], para comparar seu desempenho com a rede ConvLSTM e outra opção de algoritmo apresentada em [Zhou et al. 2020].

Uma grande variedade de aplicações distintas de CFD podem ser beneficiadas com o tratamento de aprendizagem profunda. Nas análises de eólica offshore, em que as condições de vento reais e sazonais devem ser avaliadas, pretende-se avaliar a metodologia de aprendizado profundo na previsão do potencial de geração, como apresentado em [Avila et al. 2017]. Em geral, espera-se que a metodologia de aprendizado profundo, quando aplicada aos modelos de alta turbulência permitam uma redução no custo computacional.

## References

- Aigner, S. and Körner, M. (2019). Futuregan: Anticipating the future frames of video sequences using spatio-temporal 3d convolutions in progressively growing gans. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-2/W16:3–11.
- Avila, M., Gargallo-Peiró, A., and Folch, A. (2017). A cfd framework for offshore and onshore wind farm simulation. In *Journal of Physics: Conference Series*, volume 854, page 012002. IOP Publishing.
- Bhatnagar, S., Afshar, Y., Pan, S., Duraisamy, K., and Kaushik, S. (2019). Prediction of aerodynamic flow fields using convolutional neural networks. *Computational Mechanics*, 64:525–545.
- Kochkov, D., Smith, J. A., Alieva, A., Wang, Q., Brenner, M. P., and Hoyer, S. (2021). Machine learning–accelerated computational fluid dynamics. *Proceedings of the National Academy of Sciences*, 118(21):e2101784118.
- Mathias, M. S., de Almeida, W. P., Coelho, J. F., de Freitas, L. P., Moreno, F. M., Netto, C. F. D., Cozman, F. G., Reali Costa, A. H., Tannuri, E. A., Gomi, E. S., and Dottori, M. (2022a). Augmenting a physics-informed neural network for the 2d burgers equation by addition of solution data points. In Xavier-Junior, J. C. and Rios, R. A., editors, *Intelligent Systems*, pages 388–401. Springer International Publishing.
- Mathias, M. S., de Barros, M. R., Coelho, J. F., de Freitas, L. P., Moreno, F. M., Netto, C. F. D., Cozman, F. G., Costa, A. H. R., Tannuri, E. A., Gomi, E. S., and Dottori, M. (2022b). A physics-informed neural network to model port channels.

- Ozaki, H. and Aoyagi, T. (2022). Prediction of steady flows passing fixed cylinders using deep learning. *Scientific Reports* 2022 12:1, 12:1–12.
- Shi, X., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W.-K., and Woo, W.-c. (2015). Convolutional lstm network: A machine learning approach for precipitation nowcasting. *Advances in neural information processing systems*, 28.
- Sirignano, J., MacArt, J. F., and Freund, J. B. (2019). Dpm: A deep learning pde augmentation method (with application to large-eddy simulation). *Journal of Computational Physics*, 423.
- Usman, A., Rafiq, M., Saeed, M., Nauman, A., Almqvist, A., and Liwicki, M. (2021). Machine learning computational fluid dynamics. In *2021 Swedish Artificial Intelligence Society Workshop (SAIS)*, pages 1–4.
- Vinuesa, R. and Brunton, S. L. (2022a). Enhancing computational fluid dynamics with machine learning. *Nature Computational Science* 2022 2:6, 2:358–366.
- Vinuesa, R. and Brunton, S. L. (2022b). Enhancing computational fluid dynamics with machine learning. *Nature Computational Science*, 2(6):358–366.
- Wang, Z., Bovik, A., Sheikh, H., and Simoncelli, E. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612.
- Ye, S., Zhang, Z., Song, X., Wang, Y., Chen, Y., and Huang, C. (2020). A flow feature detection method for modeling pressure distribution around a cylinder in non-uniform flows by using a convolutional neural network. *Scientific Reports*, 10:4459–4459.
- Yeung, P. K., Zhai, X. M., and Sreenivasan, K. R. (2015). Extreme events in computational turbulence. *Proceedings of the National Academy of Sciences of the United States of America*, 112:12633–12638.
- Zhou, Y., Dong, H., and El Saddik, A. (2020). Deep learning in next-frame prediction: A benchmark review. *IEEE Access*, 8:69273–69283.