# Master project - Inverse Design of Gas-Bearings Supported Turbocompressors using Generative Adversarial Networks

November 24, 2022

## General information

- Laboratory: Laboratory of Applied Mechanical Design (LAMD)
- Supervisor: Soheyl Massoudi, Prof. J. Schiffmann
- Location: Lausanne or Neuchâtel (travel allowance offered)
- Starting date: ASAP
- Duration: Semester
- Contact: soheyl.massoudi@epfl.ch
- Keywords: Generative Adversarial Networks (GANs), cGANs, Python, Tensorflow/Pytorch, Turbomachinery, Gas Bearings, Herringbone Grooved Journal Bearings

## Background

Generative Adversarial Networks (GANs) have been introduced in 2014 [1]. They are often made of two competing neural networks. A generator $G$ aims to create outputs that resemble the original data while a discriminator $D$ that distinguishes between real and fake samples. As such, they can be used to mimic art, create photo realistic portraits of human beings as seen on Figure 1 [2] or transfer art style from one painting to another [3]. Coupled with a Natural Language Processing model, GANs can also generate images from text.

Engineering design requires synthesizing new designs that meet the desired performance requirements. A conventional design process involves iterative optimization and performance evaluation. A variant of GANs, called conditional GANs or cGANs, enables the direct design synthesis for given

target performance [4, 5]. GANs can find high-quality designs outide the training range[6], initialize an optimization with a distribution of designs close to the the Pareto distribution, or bypass the optimization process altogether [7].

Figure 1: A fake portrait by the Nvidia StyleGAN generator.

## Objective

The objective is the creation of state-of-the-art conditional GANs for optimal gas-bearings supported turbocompressor designs. The trained generator will then be able to emulate high-quality designs with respect to the select performance metrics such as stability, load capacity, efficiency, or robustness.

## Tasks (working plan guideline)

1. Literature review on GANs, tensorflow library and good Python coding practices
2. Build a Python framework to create, train and exploit GANs. Choose to work with GANs, cGANs or both
3. Use the latent space and/or conditions to manipulate the generator of the GAN towards creating wanted performing turbocompressor designs
4. Improve your framework by implementing the state-of-the art of GANs for design
5. Generalize your framework to different optimization objectives and rotor layouts

## References


[1] Ian Goodfellow et al. "Generative adversarial networks". In: *Communications of the ACM* 63.11 (2020), pp. 139–144.

[2] Tero Karras et al. "Analyzing and improving the image quality of stylegan". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 8110–8119.

[3] Jun-Yan Zhu et al. "Unpaired image-to-image translation using cycle-consistent adversarial networks". In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2223–2232.

[4] Mehdi Mirza and Simon Osindero. "Conditional generative adversarial nets". In: *arXiv preprint arXiv:1411.1784* (2014).

[5] Amin Heyrani Nobari, Wei Chen, and Faez Ahmed. "Pcdgan: A continuous conditional diverse generative adversarial network for inverse design". In: *arXiv preprint arXiv:2106.03620* (2021).

[6] Wei Chen and Faez Ahmed. "Padgan: Learning to generate high-quality novel designs". In: *Journal of Mechanical Design* 143.3 (2021).

[7] Wei Chen, Kevin Chiu, and Mark D Fuge. "Airfoil design parameterization and optimization using bézier generative adversarial networks". In: *AIAA journal* 58.11 (2020), pp. 4723–4735.