



# 《计算机图形学基础》

## 习题课4

助教 李相利

2022年04月29日



# 计图 (Jittor) 介绍



# Jittor介绍

- Jittor, 中文名为“计图”, 发布于2020年3月20日。
- 是一个基于**元算子**和**统一计算图**的深度学习框架。
- 在多种应用中, 相比国际主流平台, 性能得到大幅提升。
- 目前支持: 30+骨干网络, 27种主流GAN模型, 9种图像实例分割模型, 5种点云网络模型, 8种检测模型, 提供可微渲染功能, ...

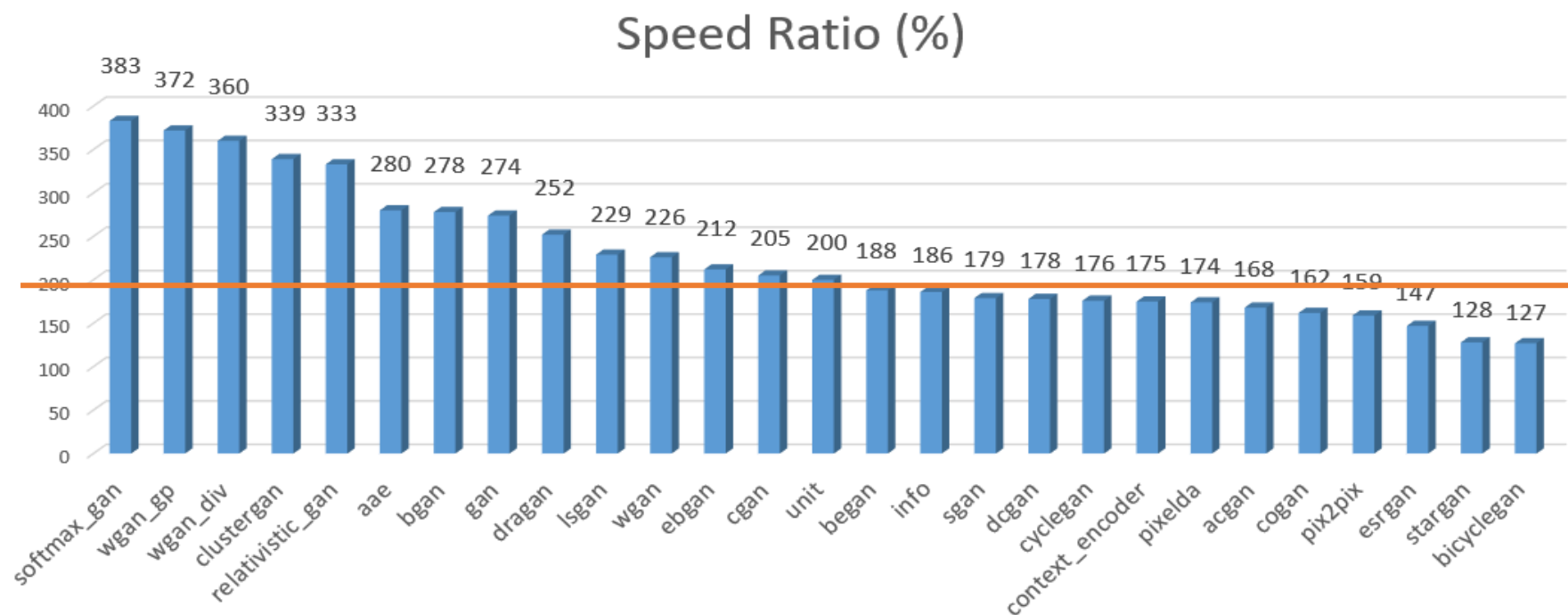


**Jittor 计图**



# Jittor介绍

- 以GAN模型库举例：
  - 所有模型速度比PyTorch均得到提升
  - 半数模型性能提升超过一倍





# Jittor安装与使用

Linux 安装

OS	Ubuntu / Debian / 统信 / ...	CentOS / redhat / 麒麟 / ...	Mac	Windows
Package	Conda	Pip	Source	
Platform	CPU	CUDA	OpenCL	
Install	<pre>sudo apt install python3.7-dev libomp-dev python3.7 -m pip install jittor python3.7 -m jittor.test.test_example</pre>			

<https://cg.cs.tsinghua.edu.cn/jittor/download/>



# Jittor安装与使用

Windows 安装

OS	Ubuntu / Debian / 统信 / ...	CentOS / redhat / 麒麟 / ...	Mac	Windows
Package	Conda	Pip	Source	
Platform	CPU	CUDA	OpenCL	
Install	<pre># 检查python版本大于等于3.8 python --version python -m pip install jittor python -m jittor.test.test_core python -m jittor.test.test_example python -m jittor.test.test_cudnn_op</pre>			

<https://cg.cs.tsinghua.edu.cn/jittor/download/>



# Jittor安装与使用

## ■ 在线浏览:

<https://nbviewer.jupyter.org/github/Jittor/LearnJittorBasicIn60Min/tree/master/>



LearnJittorBasicIn60Min

### Name

◀ Jittor's repositories

📄 计图入门教程 0 --- 介绍与安装.ipynb

📄 计图入门教程 1 --- 基本概念.ipynb

📄 计图入门教程 2 --- 如何训练一个简单线性回归.ipynb

📄 计图入门教程 3 --- 尝试解决一个实际问题.ipynb

🔗 README.md

🔗 mnist.png

代码演示



# Jittor安装与使用

<https://cg.cs.tsinghua.edu.cn/jittor/tutorial/>

## 2020年12月教程

- [Jittor可微渲染库使用教程](#)

## 2020年10月教程

- [Jittor点云模型库使用教程](#)
- [Jittor实例分割模型库使用教程](#)

## 2020年5月教程

- 计图研讨会视频: [GAN模型库的使用与可视化](#)
- 计图研讨会视频: [如何使用计图提升AI研究](#)
- [使用Jittor实现Conditional GAN](#)
- [Windows/Mac/Linux通过Docker安装计图](#)

- [模型辅助转换工具教程](#)
- [计图MPI多卡分布式教程](#)
- [利用Code算子简化高性能自定义算子开发](#)

## 2020年3月教程

- 基本概念: [Op, Var](#)
- 元算子: [通过元算子实现自己的卷积层](#)
- 自定义算子: [使用C++和CUDA编写您的算子, 并其进行即时编译](#)
- 示例1: [线性回归](#)
- 示例2: [MNIST图像分类](#)
- 示例3: [图像生成之 LSGAN](#)
- 示例4: [语义分割之 DeepLabV3+](#)
- 示例5: [目标检测之 SSD](#)





# Jittor安装与使用

## ■ 我们提供了从PyTorch到Jittor的模型迁移工具

```
from torch import nn

class Linear(nn.Module):
    def __init__(self, n_in, n_out):
        self.w = random(n_in, n_out)
        self.b = zeros(n_out)

    def forward(self, x):
        return matmul(x, self.w) + self.b

class Model(nn.Module):
    def __init__(self):
        self.linear1 = Linear(1, 64)
        self.linear1 = Linear(64, 128)
        self.linear1 = Linear(128, 1)

    def forward(self, x):
        x = self.linear1(x)
        x = relu(x)
        x = self.linear2(x)
        x = relu(x)
        return self.linear3(x)
```

辅助转换脚本

```
import jittor as jt

class Linear(jt.Module):
    def __init__(self, n_in, n_out):
        self.w = random(n_in, n_out)
        self.b = zeros(n_out)

    def execute(self, x):
        return matmul(x, self.w) + self.b

class Model(jt.Module):
    def __init__(self):
        self.linear1 = Linear(1, 64)
        self.linear1 = Linear(64, 128)
        self.linear1 = Linear(128, 1)

    def execute(self, x):
        x = self.linear1(x)
        x = relu(x)
        x = self.linear2(x)
        x = relu(x)
        return self.linear3(x)
```



# Jittor安装与使用

## ■ 计图支持的骨干网络一览

```
from jittor.models import \
    resnet152, \
    res2net, \
    alexnet, \
    googlenet, \
    inception_v3, \
    mnasnet, \
    mobilenet_v2, \
    resnext101_32x8d, \
    shufflenetv2, \
    squeezenet, \
    wide_resnet101_2
```

## 参考资料:

- [模型辅助转换工具教程](#)
- [计图模型文档](#)

更多的模型和应用在不断  
完善当中，尽情期待



# Jittor安装与使用

<https://cg.cs.tsinghua.edu.cn/jittor/resources/>

动态

论文

资源

感谢大家对 Jittor 的关注, 下面总结 Jittor 的一些优质资源. 如有疑问, 可以在 [Github](#) 提 Issue , 或者通过 Email: [jittorqq@qq.com](mailto:jittorqq@qq.com) 联系我们. 欢迎大家使用 Jittor 进行科学研究工作.

### 计图可微渲染库

可微渲染是近些年计算机视觉领域逐渐兴起的方向之一, 通过计算渲染过程的导数, 使得从单张图片学习三维结构逐渐成为现实. 可微渲染目前被广泛应用于三维重建, 同时在人体重建、人脸重建、三维属性估计等应用. 目前, Jittor 已经支持了两种主流的可微渲染器 NMMR 和 SoIRas. 我们对这两个渲染器与 PyTorch 进行了渲染速度的对比, 渲染速度分别是 PyTorch 的 10.59 倍和 4.09 倍. 我们为大家提供了 8 个教程对 Jittor 进行熟悉和学习, 其中包括 6 个基础教程和 2 个进阶教程. 上面展示了部分渲染结果.

Code

### 计图点云模型库

点云数据的获取与处理是图形学和三维视觉中的重要问题, 在测绘、自动驾驶等方面有着广泛的应用. 由于其巨大的应用前景, 近年来, 点云处理引起了相关技术人员广泛关注, 涌现了一批非常出色的工作. 计图框架本次发布的点云模型库包括几种最有影响力的模型: PointNet, PointNet++, PointCNN, DGCRN 和 PointCnn, 支持分类和分割. PointNet++ 和 PointCNN 模型训练速度比 PyTorch 提升一倍以上.

Code

### 计图语义分割模型库

语义分割是计算机视觉中的基础任务之一. 如图所示, 其目标是从像素的角度来理解图片, 对原图中的每个像素都进行分类标注. 语义分割是众多算法和人脸解析、视频语义分割等算法的基础. 同时, 语义分割在无人驾驶、智能编码、智能农业、虚拟现实等方面有着广泛的应用. 继5月30日发布GAN模型库之后, 计图平台进一步发布语义分割模型库, 加强对视觉任务的支持. 目前, Jittor 已经支持了目前主流的语义分割算法. 其中包含了三种经典的 Backbone, 以及六种常见的分割模型. 所发布模型的训练速度相比于 PyTorch 均有所提升.

Code

### 计图实例分割模型库

Jittor 实例分割模型库一共包含了 6 种 Backbone, 和 11 类检测分割模型, 包含最经典的 Mask RCNN 系列, 实时实例分割网络以及人体分割网络等等. 经过我们多次测试, 所有模型的推理速度都比 PyTorch 都要快, 其中最快的比 PyTorch 快 30% 左右. 使用我们的实例分割模型库, 不仅可以方便的使用多类模型作为 Baseline, 而且正是由于 Jittor 的高性能, 能够使研究效率有较大的提升.

Code

### 计图GAN模型库

Jittor GAN 模型库一共包括了从 2014 到 2019 最主流的 27 种 GAN 模型. 这 27 个 GAN 总计被引用 60903 次, 每篇文章平均引用次数为 2276. 我们经过严谨的测试看出, 所有 Jittor 模型的训练速度都比 PyTorch 快. 在这些模型中, 最快的是 PyTorch 的 3.83 倍, 最慢的是 PyTorch 的 1.27 倍, 平均为 PyTorch 的 2.26 倍. 这意味着假设 PyTorch 平均需要训练 100 个小时, 那么 Jittor 只需训练 50.04 (100 / 2.26) 个小时. 因此, 使用我们的模型库可以大大提高研究人员的研究效率.

Code



# Jittor安装与使用

<https://cg.cs.tsinghua.edu.cn/jittor/assets/docs/index.html>

🏠 Jittor

Search docs

内容一览:

Jittor: 即时编译深度学习框架

模块API:

- jittor
- jittor.nn
- jittor.models
- jittor.optim
- jittor.init
- jittor.contrib
- jittor.dataset
- jittor.transform
- jittor.mpi
- jittor.linalg

其他:

教程

TODO

🏠 » 欢迎查阅计图文档

## 欢迎查阅计图文档

内容一览:

- Jittor: 即时编译深度学习框架
  - 大纲
  - 快速开始
  - 安装
  - Pip 安装
  - 一键脚本安装
  - 手动安装
  - 教程
  - 贡献
  - 联系我们
  - 团队
  - 引用
  - 版权声明

## 模块API:



### Jittor开发者交流群

群号: 761222083



扫一扫二维码，加入群聊。

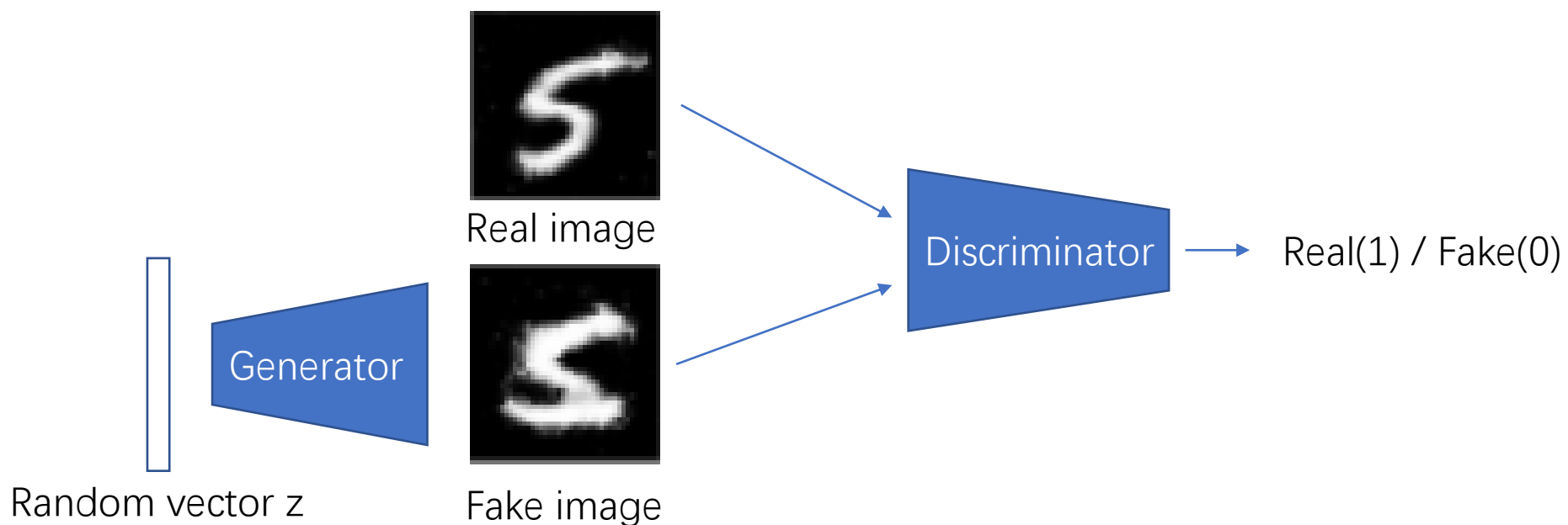
 QQ



# Conditional GAN

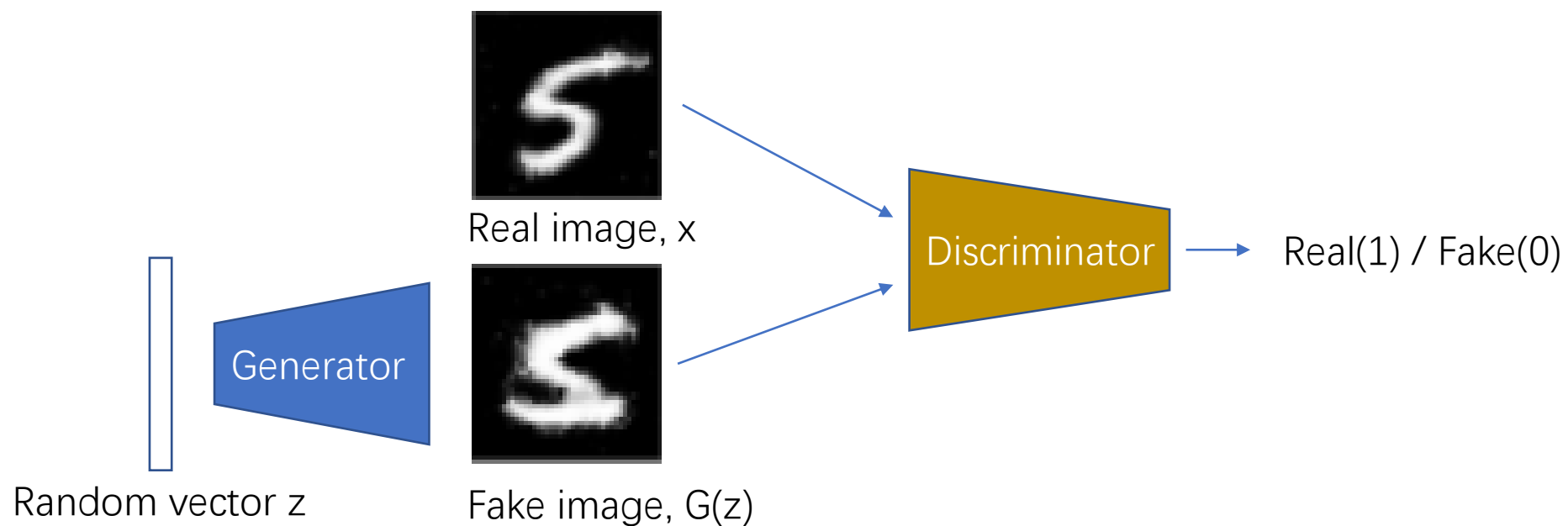
# GAN

- 生成式对抗网络 (GAN, Generative Adversarial Networks)
- 一般由生成器 (Generator) 和判别器 (Discriminator) 两部分组成。



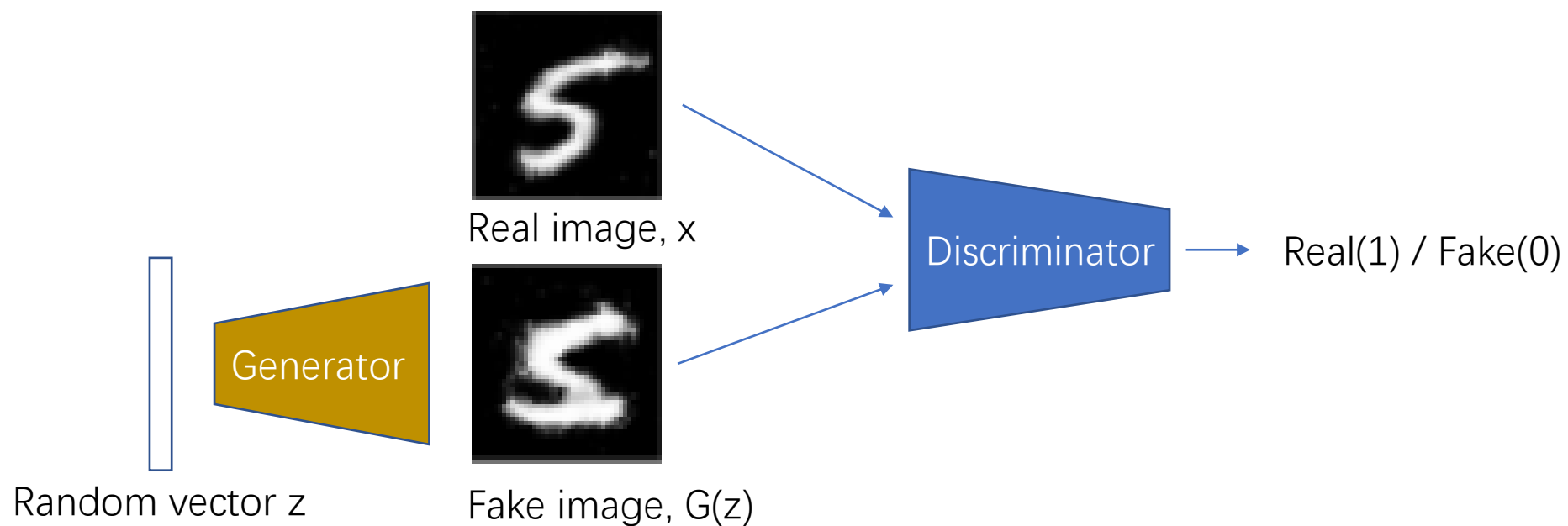
# GAN

- 分为两个部分进行训练。
- Discriminator:  $L_D = (D(G(z))^2 + (1 - D(x))^2)/2$



# GAN

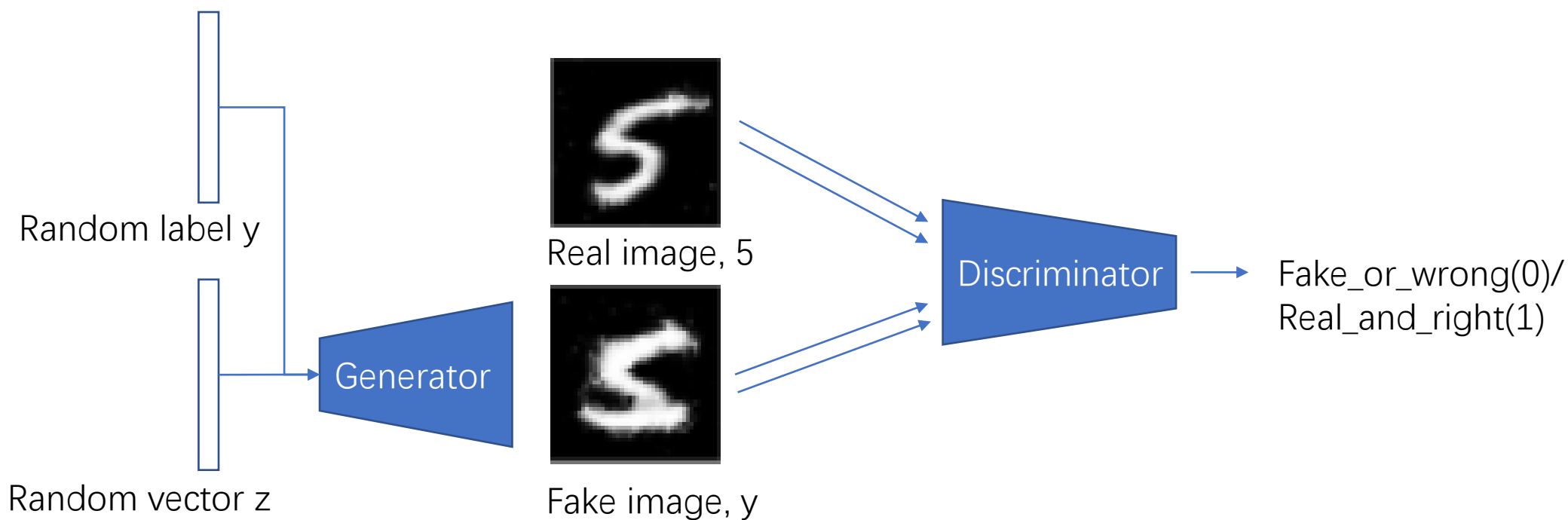
- 分为两个部分进行训练。
- Generator:  $L_G = (1 - D(G(z)))^2$





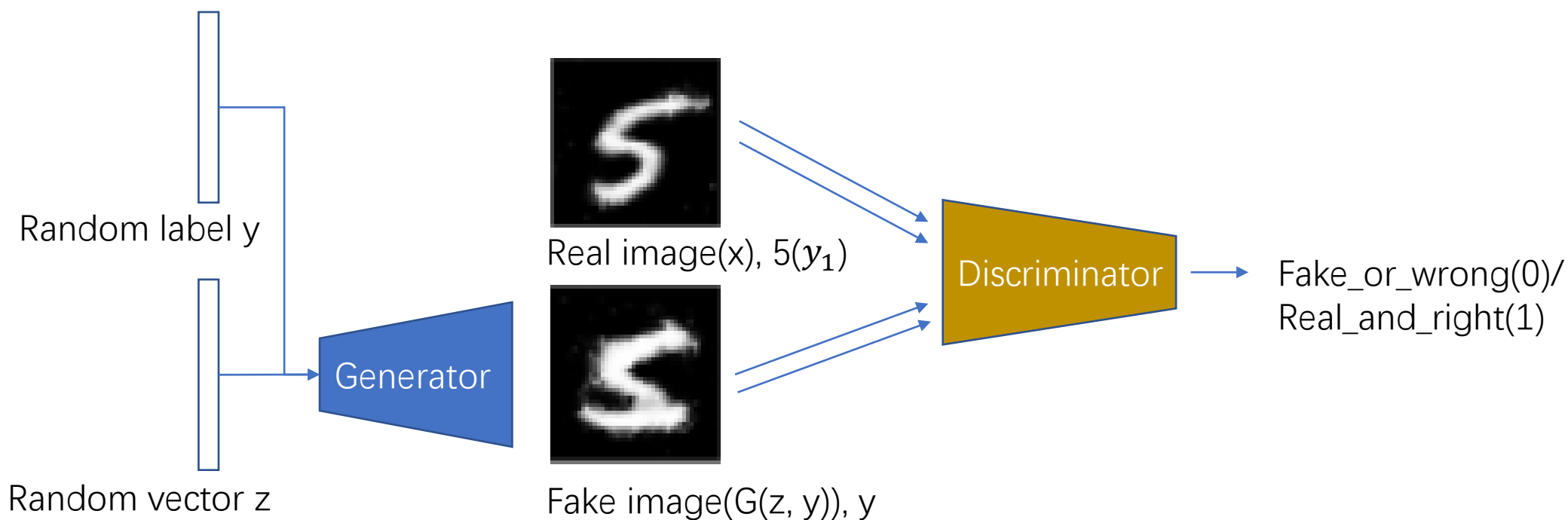
# Conditional GAN

- 在GAN的基础上加上了对输出的控制。



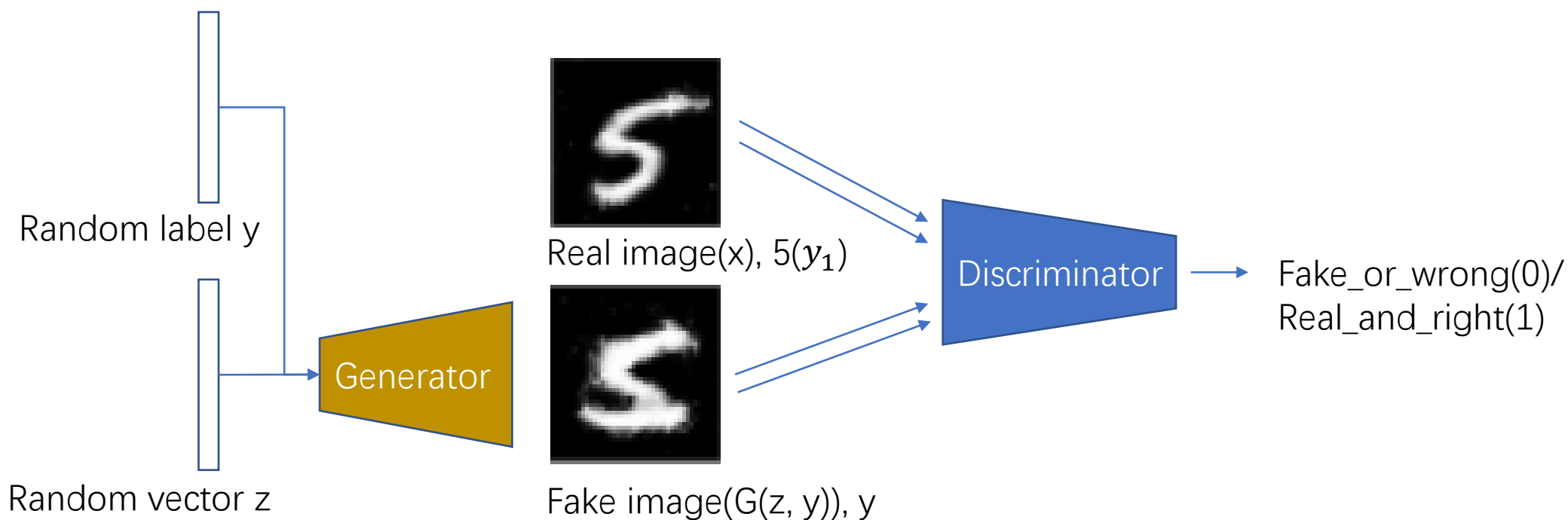
# Conditional GAN

- Discriminator:  $L_D = (D(G(z, y), y))^2 + (1 - D(x, y_1))^2)/2$



# Conditional GAN

- Generator:  $L_G = (1 - D(G(z, y), y))^2$





# 提交要求

- 本次作业要求在头歌平台上进行评测
- 评测地址: <https://www.educoder.net/competitions/index/Jittor-3>
- 请同学们注册账号, 按平台中的要求完成评测和开源
- 并在网络学堂提交实验报告 (report.pdf)



# 提交要求

- 实验报告应包含以下5个部分：
  - 1. 最终生成的图片（result.png）。
  - 2. 头歌平台通过评测的截图，如图。
  - 3. 头歌平台通过评测的评测编号，如图中第一列的两行数字，2022041810215376658720。
  - 4. Gitlink项目开源地址。
  - 5. Gitlink项目主页截图。



2022041810215  
376658720

submit.zip...

杨国辉

2022-04-18  
10:21:53

完成

0.9944

是否通过: 1



# 开源要求

- 为了让同学们熟悉开源项目的创建和管理，本次作业要求在Gitlink中开源
  - [www.gitlink.org.cn](http://www.gitlink.org.cn)。
- 一个好的开源项目应包含合适的项目名称、项目简介、README文件、gitignore文件以及项目的源码文件。
- 开源的项目名和项目简介中应包含所使用的框架，方便其他用户检索：
  - 项目名：CGAN\_jittor
  - 项目简介：A Jittor implementation of Conditional GAN (CGAN).
  - 项目语言选择Jittor



# 开源要求

- README文件的主要功能是对项目的功能和使用方式进行介绍，其书写规则可以参考：
  - <https://chinese.freecodecamp.org/news/how-to-write-a-good-readme-file/>
- gitignore文件的主要功能是屏蔽某些文件，使得这些文件不被追踪（tracked），push后也不会上传到gitlink等平台，一般用于屏蔽有代码自动生成的（编译结果文件、中间结果文件等），或占用空间较大的文件（一般数据集等文件会存放在网盘，然后在README中提供下载链接），其书写规则可以参考：
  - <https://git-scm.com/docs/gitignore>。



*Jittor* 计图

课程作业要求汇总：

<https://cloud.tsinghua.edu.cn/d/46dfd7381cf245e7b265/>

谢谢！