

# manim 常见问题 v2.1

鹤翔万里 & catfish

2020 年 2 月 24 日

## 目录

<b>1</b>	<b>安装问题</b>	<b>3</b>
1.1	Python 问题	3
<b>2</b>	<b>运行时问题</b>	<b>4</b>
2.1	import 问题	4
2.2	L <sup>A</sup> T <sub>E</sub> X 问题	4
2.3	中文显示问题	5
2.4	文字问题	5
2.5	素材引用问题	7
<b>3</b>	<b>其它问题</b>	<b>8</b>
Q1:	有什么 manim 教程	8
Q2:	没有 manim 源码	8
Q3:	群友用的 manim 都是什么版本	8
Q4:	如何使用傅里叶级数作图	8
Q5:	svg 用什么软件制作	8
Q6:	一些比较复杂, 操纵东西比较多的动画怎么做	8
Q7:	一个 self.play 里写两个 ApplyMethod 只对一个起作用怎么办	8
Q8:	动画怎么显示旋转一个物体	8
Q9:	怎么控制物体移动或者 Transform 的加速度	8
Q10:	数学符号/公式用 L <sup>A</sup> T <sub>E</sub> X 怎么打	9
Q11:	使用一些特殊 L <sup>A</sup> T <sub>E</sub> X 的外部包	9
Q12:	使用 L <sup>A</sup> T <sub>E</sub> X 外部包, 编译错误或者无显示	9
Q13:	如何解决二维画面中的图层问题	11
Q14:	如何导出 gif 文件	12
Q15:	如何导出透明的图片或者视频	12
Q16:	渲染视频的画质和帧率怎么调整	12
Q17:	有没有什么好的场景例子供学习	13
<b>4</b>	<b>注意</b>	<b>14</b>

# 1 安装问题

安装时最好不要看 README.md 自己研究。推荐一视数学卷毛杨的两个教程

- <https://www.bilibili.com/video/av38126904>
- <https://www.bilibili.com/read/cv4139851>

## 1.1 Python 问题

**Q1: 使用 anaconda, 命令行输入 python 无反应或报错**

考虑 path 环境变量是否填全<sup>1</sup>, path 变量里应该有

```
<your_path>\Anaconda3;  
<your_path>\Anaconda3\Scripts;  
<your_path>\Anaconda3\Library\bin;
```

**Q2: pip install ... 时满屏红字报错, 或者安装过慢**

更换国内镜像源, 使用

```
pip install -r requirements.txt -i https://pypi.tuna.tsinghua.edu.cn/simple
```

代替<sup>2</sup>

```
pip install -r requirements.txt
```

**Q3: pip 安装 pycairo 总是失败**

下载 pycairo 对应版本的 whl 包<sup>3</sup>

```
pycairo-1.18.2-cp37-cp37m-win_amd64.whl
```

并手动安装

```
pip install pycairo.....whl
```

**Q4: pip 安装过包, 但运行时提示没有模块**

考虑电脑上是否有多个 Python, 确定 pip 把包装到了哪个 Python 上面

---

<sup>1</sup>安装 anaconda 时是否勾选添加到 path 变量

<sup>2</sup>临时换源

<sup>3</sup>群文件中有, 注意 python 版本和系统版本是否均合适

## 2 运行时问题

### 2.1 import 问题

**Q1:** 没有模块 `big_ol_pile_of_manim_imports`

将文件中的

```
from big_ol_pile_of_manim_imports import *
```

改成

```
from manimlib.imports import *
```

### 2.2 L<sup>A</sup>T<sub>E</sub>X 问题

**Q1:** 报错 `Latex error converting to dvi`

先不要管错误在哪，先把 `manimlib/constants.py` 中的 `TEX_USE_CTEX` 改成 `True` 再运行

**Q2:** 报错 `xelatex error converting to xdv`

若为 Windows 系统，先把 `manimlib/constants.py` 的第 29 行

```
MEDIA_DIR = "./media"
```

改成

```
MEDIA_DIR = os.path.join(os.getcwd(), "media")
```

再进行尝试

#### I. 若安装的 T<sub>E</sub>X 发行版为 MiK<sub>T</sub>E<sub>X</sub>

1. MiK<sub>T</sub>E<sub>X</sub> 的有关路径是否添加到环境变量中
2. 是否有包没有装全

对于 2.，可以正常运行一遍 `WriteStuff` 场景，看是否又框弹出提示 `install` 什么东西，如果有，则 `install`，并重复运行安装运行安装... 直到不报错为止。或者使用 `tex` 编辑器（`TEXStudio`）并使用 `xelatex` 手动编译 `media/TeX` 文件夹中的 `.tex` 文件，查看是否有包没有安装

对于没有 1.、2. 问题却依旧报错的，可以选择重新安装新版 MiK<sub>T</sub>E<sub>X</sub> 或者安装 `TEXLive-full` 版

#### II. 若安装的 T<sub>E</sub>X 发行版为 T<sub>E</sub>XLive

1. T<sub>E</sub>XLive 有关路径是否添加到环境变量中
2. 安装的是否为 full 版本

#### III. 若安装的 T<sub>E</sub>X 发行版不为以上两款

建议换成 `TEXLive-full` 版或者 `MiKTEX`，并且在重新安装前请删除旧版

**Q3: 报错在文件夹内找不到 svg 文件**

清空 media/TeX 文件夹内全部内容，再次运行带文字的场景，查看 TeX 文件夹中的内容

- I. 若仅有 tex 文件和 log 文件, 按照 2.2 中方法处理
- II. 若含有 xdv 文件但没有 svg 文件
  1. dvisvgm 是否添加到环境变量, 可以使用 `dvisvgm --version` 观察是否由报错来检查
  2. dvisvgm 版本是否过低, 若 `dvisvgm --verison` 的输出版本号为 1 开头, 请更换新版 dvisvgm<sup>4</sup>, 并注意将含有 dvisvgm 的文件夹添加到环境变量中

**2.3 中文显示问题****Q1: 含有中文的 TextMobject 编译报错 Latex error converting to dvi**

将 manimlib/constants.py 中的 `TEX_USE_CTEX` 改成 True 再尝试

**Q2: 英文可以正常显示，中文不报错，但不显示**

考虑使用的是否为 `TextMobject` 而不是 `TexMobject`

**2.4 文字问题****Q1: TextMobject 和 TexMobject 有什么区别**

`TextMobject` 和 `TexMobject` 使用的都是  $\LaTeX$  语法  
 其中 `TextMobject` 文字模式相当于直接在  $\LaTeX$  环境下书写  
`TexMobject` 公式模式使用的是  $\LaTeX$  的 `\begin{align*}` 环境或者可以看成加了 `$$` 的环境  
 使用 `TextMobject` 与 `TexMobject` 书写公式时

$$\text{TextMobject}(\text{"文字\$公式\$"}) \iff \text{TexMobject}(\text{"\\text{文字}公式"})$$
**Q2: TextMobject 中怎么改字体**

`TextMobject` 中只能使用  $\LaTeX$  的内置字体族和字体形状，包括：

- 罗马字体 `\textrm{textrm}`
- 无衬线字体 `\textsf{textsf}`
- 打字机字体 `\texttt{texttt}`
- 直立形状 `\textup{textup}`
- 意大利形状 `\textit{textit}`
- 倾斜形状 `\textsl{textsl}`
- 小型大写 `\textsc{TEXTSC}`

---

<sup>4</sup>上网下载、或者使用群文件中的版本

**Q3: 想自定义字体怎么办**

使用新版 manim 特有的 `Text()` 类, 方法如下 `Text("文字", font="字体")`, 其中字体要填写在计算机内存储的格式<sup>5</sup>, 但是不能使用  $\text{\LaTeX}$  语法书写公式

**Q4: 想用自定义字体写公式怎么办**

可以使用群文件里 cigar666 编写的 `MyText()` 类 Cigar 牛逼

**Q5: TexObject 中换行是什么**

四个右划线 `\\`, Python 转义右划线, 所以涉及到 `\` 的均要写成两个 `\\`, 而换行在  $\text{\LaTeX}$  中是两个右划线, 所以要写成四个<sup>6</sup>

**Q6: 公式怎么对齐**

- I. 直接在 `TexObject` 中使用 `&` 对齐
- II. 两个 `mobject` 对齐, 使用 `obj2.next_to(obj1, DOWN, aligned_edge=LEFT)` 使 `obj2` 在 `obj1` 下方, 并且左对齐
- III. `VGroup` 内对齐, 使用 `group.arrange(DOWN, aligned_edge=LEFT)` 使 `VGroup` 中的子元素依次向下排开, 并左对齐

写公式的示例:

[https://github.com/Elteoremadebeethoven/AnimationsWithManim/blob/master/English/3\\_text\\_like\\_arrays/scenes.md](https://github.com/Elteoremadebeethoven/AnimationsWithManim/blob/master/English/3_text_like_arrays/scenes.md)

**Q7: TexObject 上色问题的处理办法**

- I. 将上色的字符分开, 使用 `text[i].set_color(color)` 来上色
- II. 将上色的字符分开, 使用 `text.set_color_by_tex_to_color_map(t2c)` 传入 `t2c` 字典来对相同的字符串上色
- III. 只传入一个字符串, 但同时传入 `tex_to_color_map=t2c` 来自动拆分上色 (容易出问题)
- IV. 只传入一个字符串, 使用 `text[0][i]` 来对细小的路径上色 (一般是一个字符一个下标)

**Q8: TexObject 的下标怎么分析**

创建函数

```
def debugTeX(self, texm):
    for i, j in zip(range(100), texm):
        tex_id = TextMobject(str(i)).scale(0.3).set_color(PURPLE)
        tex_id.move_to(j)
        self.add(tex_id)
```

在使用时先 `self.add(tex)` 然后再 `debugTeX(self, tex)`, 导出最后一帧<sup>7</sup> 观察每段字符上的标号, 即为下标

<sup>5</sup>例如 Microsoft YaHei, Source Han Sans CN

<sup>6</sup>或者在字符串前加 `r`, 正常书写

<sup>7</sup>-s 选项

**Q9: TexObject 使用 \frac 拆分时出错**

这个是 Grant 写 `tex_file_writing.py` 的一个 bug, 建议使用 `{分子 \over 分母}` 来代替 `\frac{分子}{分母}`

**2.5 素材引用问题****Q1: 使用 SVGObject 找不到 svg 文件**

- I. 直接使用绝对路径引用 svg 文件
- II. 将 svg 文件放到 `assets/svg_images/` 文件夹中

**Q2: 如何使用 jpg 或者 png 文件**

- I. 直接使用绝对路径引用, 并使用 `ImageObject`
- II. 将 jpg/png 文件放到 `assets/raster_images/` 文件夹中

## 3 其它问题

### Q1: 有什么 manim 教程

1. 群主 cigar666 的 B 站专栏

- <https://www.bilibili.com/read/readlist/rl82339>

2. pdcxs 大大转载的 manim 教程

- <https://www.bilibili.com/video/av64023740>
- 源码<https://github.com/Elteoremadebeethoven/AnimationsWithManim>

3. GitHub 上 cai-hust 的中文教程

- <https://github.com/cai-hust/manim-tutorial-CN>

4. manim 源码

### Q2: 没有 manim 源码

最好不要使用 `pip install manimlib` 来装 manim, 请在 GitHub 上 clone 下来 manim 的全部内容

### Q3: 群友用的 manim 都是什么版本

manim 不看版本, 一般使用的都是最新版 code。release 里面带版本号的都可以看作旧版

### Q4: 如何使用傅里叶级数作图

套用 Grant 写好的文件

```
active_projects/diffyq/part2/fourier_series.py
active_projects/diffyq/part4/fourier_series_scenes.py
active_projects/diffyq/part4/long_fourier_series.py
```

只需要更换 svg 素材即可 (自己制作, 或者使用群里的 svg 素材)

### Q5: svg 用什么软件制作

Adobe Illustrator(简称 AI) 或者 inkscape(简称 ink)。尽量不要使用网页版编辑器

### Q6: 一些比较复杂, 操纵东西比较多的动画怎么做

使用外部剪辑软件, 例如 Adobe Premiere Pro 或者达芬奇

### Q7: 一个 self.play 里写两个 ApplyMethod 只对一个起作用怎么办

去掉 ApplyMethod

### Q8: 动画怎么显示旋转一个物体

使用 `Ratate` 和 `Rotating`, 区别在群文件中有视频



**Q9: 怎么控制物体移动或者 Transform 的加速度**

使用 `rate_func`，一些 manim 中已经定义的在群文件中有视频

**Q10: 数学符号/公式用 L<sup>A</sup>T<sub>E</sub>X 怎么打**

请见<https://www.luogu.com.cn/blog/IowaBattleship/latex-gong-shi-tai-quan>

**Q11: 使用一些特殊 L<sup>A</sup>T<sub>E</sub>X 的外部包**

如何使用 manim 画出上面的音符，或怎么使用这些包？

在 manimlib 目录下的 `ctex_template.tex` 或者 `tex_template.tex` 文件中添加外部包的名称<sup>8</sup>就拿上面的音符为例，因为是在 `harmony` 包中的，所以在 `tex` 文件中添加 `\usepackage{harmony}` 然后新建一个 `py` 文件，写入代码

```
from manimlib.imports import *
class TestHarmony(Scene):
    def construct(self):
        harmony = TextMobject(
            "\\AAcht \\AAcht \\AAcht \\AAcht \\AAcht",
            color=WHITE,
            stroke_width=1,
            stroke_opacity=1,
        )

        self.play>ShowCreation(harmony)
        self.wait()
```

运行 `py` 文件即可

**Q12: 使用 L<sup>A</sup>T<sub>E</sub>X 外部包，编译错误或者无显示**

首先，并不是所有外部包都能在 manim 中顺利使用，大多都不支持 `xelatex` 编译，所以建议使用外部包时只用 `latex` 编译<sup>9</sup>

至于有些群友常用 `TiKz` 这个外部包，也是使用 `latex` 才能编译，在 `xelatex` 用 `\draw` 会无法显示

需要修改 `tex_template.tex` 文件，修改成如下：

```
\documentclass[preview, dvisvgm]{standalone}
```

新建 `py` 文件，写入代码

```
class TestTikz(Scene):
    def construct(self):
        text = TextMobject(
            r""
            \begin{tikzpicture}
```

<sup>8</sup>修改 `TEX_USE_CTEX` 为 `True` 的，可以只在 `ctex_template.tex` 中添加

<sup>9</sup>即把 `TEX_USE_CTEX` 改为 `True`

```
        \draw (-1, 0) — (1, 0);
    \end{tikzpicture}
    """
    color=WHITE,
    stroke_width=1,
    stroke_opacity=1,
)
self.play>ShowCreation(text)
self.wait()
```

运行 py 文件即可

**Q13: 如何解决二维画面中的图层问题**

使用  $z$  轴坐标对图层进行区分是无效的

可以使用 `pdcs` 添加的 `plot_depth`, 具体更改见下图

```
3 manimlib/camera/camera.py
@@ -48,6 +48,7 @@ class Camera(object):
48 48     # round z coordinate to nearest hundredth when comparing
49 49     "z_buff_func": lambda m: np.round(m.get_center()[2], 2),
50 50     "cairo_line_width_multiple": 0.01,
51 +     "open_plot_depth": True,
51 52     }
52 53
53 54     def __init__(self, background=None, **kwargs):
@@ -207,6 +208,8 @@ def extract_mobject_family_members(
207 208         method = Mobject.family_members_with_points
208 209         else:
209 210             method = Mobject.get_family
211 +         if self.open_plot_depth:
212 +             mobjects.sort(key=lambda m: m.plot_depth)
210 213     return remove_list_redundancies(list(
211 214         it.chain(*[method(m) for m in mobjects])
212 215     ))
```

```
11 manimlib/mobject/mobject.py
@@ -34,6 +34,7 @@ class Mobject(Container):
34 34     "name": None,
35 35     "dim": 3,
36 36     "target": None,
37 +     "plot_depth": 0,
37 38     }
38 39
39 40     def __init__(self, **kwargs):
@@ -1120,6 +1121,16 @@ def throw_error_if_no_points(self):
1120 1121         caller_name = sys._getframe(1).f_code.co_name
1121 1122         raise Exception(message.format(caller_name))
1122 1123
1124 +     def set_plot_depth(self, plot_depth_value):
1125 +         self.plot_depth = plot_depth_value
1126 +
1127 +     def get_plot_depth(self):
1128 +         return self.plot_depth
1129 +
1130 +     def set_plot_depth_by_z(self):
1131 +         # to update the plot_depth by the value of Z coordinate
1132 +         z_value = self.get_center()[-1]
1133 +         self.set_plot_depth(z_value)
1123 1134
1124 1135     class Group(Mobject):
1125 1136         def __init__(self, *mobjects, **kwargs):
```

`plot_depth` 的值越大, 运行出来的物体就越在上面

**Q14: 如何导出 gif 文件**

在新版本中，manim 导出 gif 已经失效，可以导出 mp4，后用 ffmpeg 转换。也可以按照下图修改源码

```

@@ -314,9 +314,17 @@ def combine_movie_files(self):
314 314         '-safe', '0',
315 315         '-i', file_list,
316 316         '-loglevel', 'error',
317 +     ]
318 +     if not self.save_as_gif:
319 +         commands += [
317 320         '-c', 'copy',
318 321         movie_file_path
319 322     ]
323 +     if self.save_as_gif:
324 +         movie_file_path = self.gif_file_path
325 +         commands += [
326 +             movie_file_path,
327 +         ]
320 328         if not self.includes_sound:
321 329             commands.insert(-1, '-an')
322 330

```

改过后，在输入命令时加上-i 选项，就能导出 gif 了

**Q15: 如何导出透明的图片或者视频**

在运行命令的时候加上-t 选项

- 如果是-s 保存图片，则会存储为背景透明的 png 图片
- 如果是-l/-m/-w 保存视频，则会存储为背景透明的 mov 视频文件，方便 pr 中的剪辑

**Q16: 渲染视频的画质和帧率怎么调整**

manim 的默认画质有四种

- -l 最低画质 480P15
- -m 中等画质 720P30
- --high\_quality<sup>10</sup> 高画质 1080P60
- -w 导出 (最高) 画质 1440P60(2K)

不加画质选项，默认使用-w 最高画质<sup>11</sup>。可以通过修改 constants.py 中对应的画面长宽和帧率来修改<sup>12</sup>

一般把-w 最高画质修改成 1080P60 (B 站支持的最高画质)

<sup>10</sup>没有缩写

<sup>11</sup>比如-p (虽然很多人把-p 当成了-w。。。)

<sup>12</sup>manimlib/constants.py 的 118 行开始

**Q17: 有没有什么好的场景例子供学习**

1. Grant 的代码<sup>13</sup>对应 3B1B 的视频，可能会有报错，需要魔改
2. 群文件里“manim 相关的 python 代码及视频结果”
3. 群里几个 B 站 up 主的 GitHub 库对应他们的代码
  - cigar666 [https://github.com/cigar666/my\\_manim\\_projects](https://github.com/cigar666/my_manim_projects)
  - 鹤翔万里 <https://github.com/Tony031218/manim-projects>
  - pdcxs <https://github.com/pdcxs/ManimProjects>
  - 有一种悲伤叫颓废 <https://github.com/136108Haumea/my-manim><sup>14</sup>

---

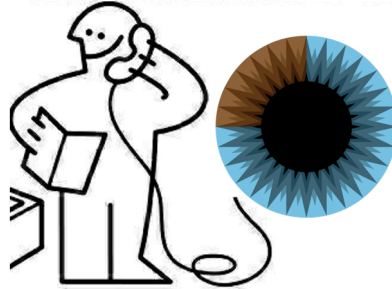
<sup>13</sup>active\_projects 和 old\_projects

<sup>14</sup>目前还是空的（颓废.jpg）

## 4 注意

如果有以上之外的问题, 可以在群里提出, 或者按照下图操作

这种情况你应该致电 格兰特·桑德森



也请注意群规第 3,4 条

- 3. 虽为 manim 交流群, 但不要一有问题就提出来, 简单的问题能自己解决最好, 不能解决时再寻求帮助
- 4. 群主和管理员平时较忙, 有时若不能及时回复敬请谅解