

ziwang-com / comfyui-fonts

<> 法典

问题

拉取请求

行动

项目

维基

安全

见解

设计

1 star

0 forks

1 watching

Branches

Activity

Tags

Public repository

1 分支

0 标签

转到文件



Go to file

添加文件

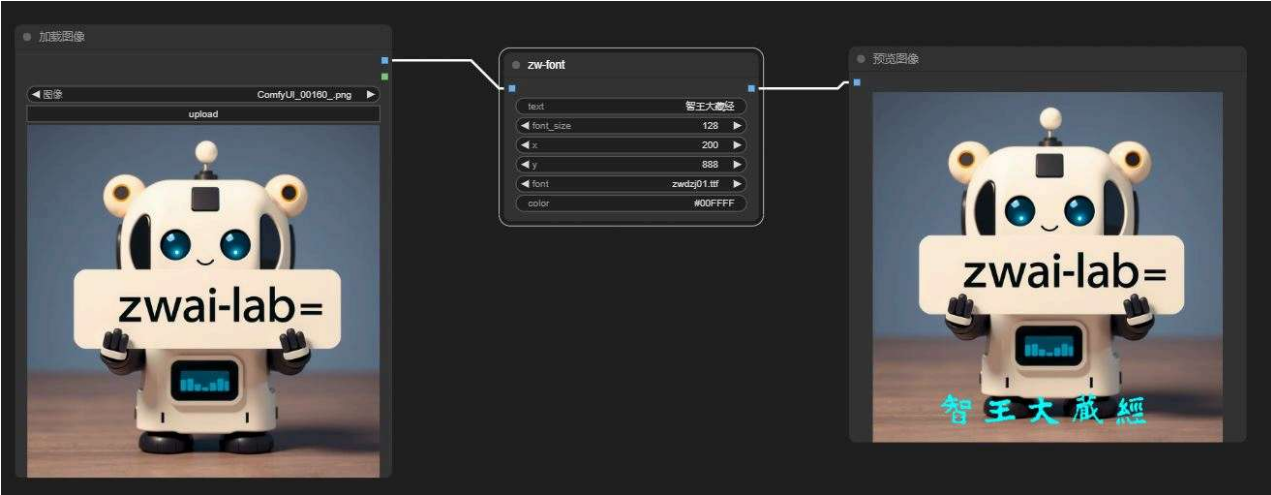
法典

 ziwang-com 网站 更新 README.md	62FA5C9 · 1 minute ago	
 资产	通过上传添加文件	19 minutes ago
 TMP	通过上传添加文件	19 minutes ago
 README.md	更新 README.md	1 minute ago
 __init__.py	通过上传添加文件	19 minutes ago
 pyproject.toml 文件	通过上传添加文件	19 minutes ago
 requirements.txt	通过上传添加文件	19 minutes ago
 zlogo64red.ico	通过上传添加文件	19 minutes ago
 zw_nodes.py	通过上传添加文件	19 minutes ago
 zw_tool.py	通过上传添加文件	19 minutes ago

自述文件

comfyUI 字体



- comfy-font万能字体节点node，支持truetype，freetype，open1等经典字库格式，也支持jpg、png等图片格式，还支持gif、短视频等各种视频格式，以及api接口。
- Comfy 字体通用字体节点，支持 Truetype、FreeType、Open1 等经典字体库格式，以及 JPG、PNG 等图片格式。还支持 GIF、短视频等多种视频格式，以及 API 接口。

安装

ComfyUI 字体

使用 ComfyUI Manager:

在 ComfyUI 管理器中，查找ComfyUI-fonts.安装它。

手动安装#1:

- 将此仓库克隆到 ComfyUI 中的文件夹custom_nodes中并安装依赖项。
- CD custom_nodes
- git clone https://github.com/ziwang-com/comfyui_fonts.git

手动安装#2:

- 在 https://github.com/ziwang-com/comfyui_fonts 下载项目打包文件。
- 解压，把解压后的目录改为：comfyui_fonts
- 手动copy到custom_nodes目录

正确目录结构为:

- 舒适UI/custom_nodes/comfyui_fonts
-

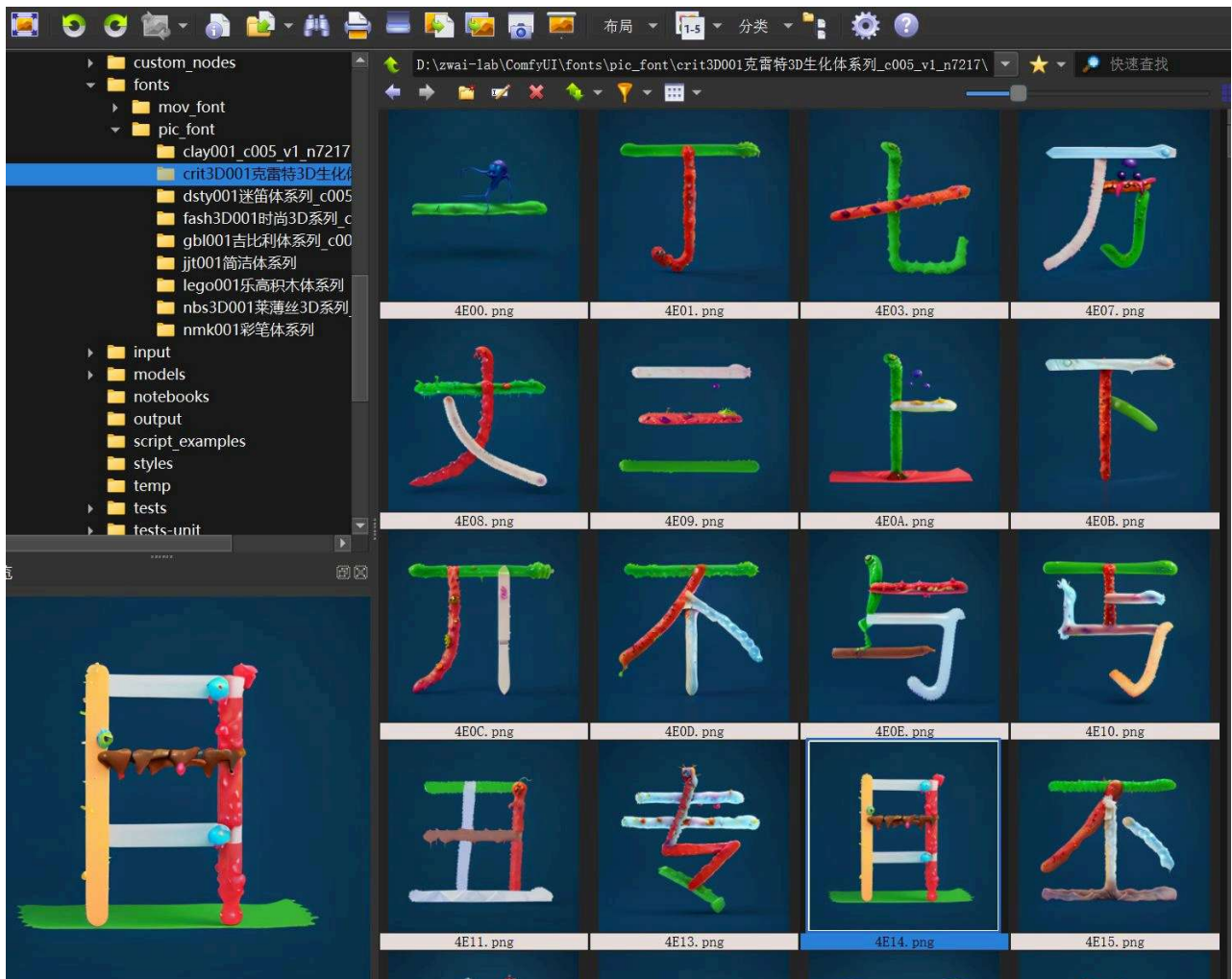
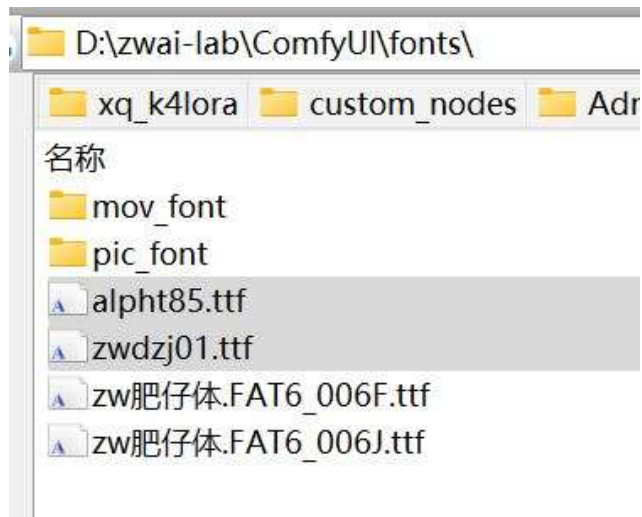
主要节点

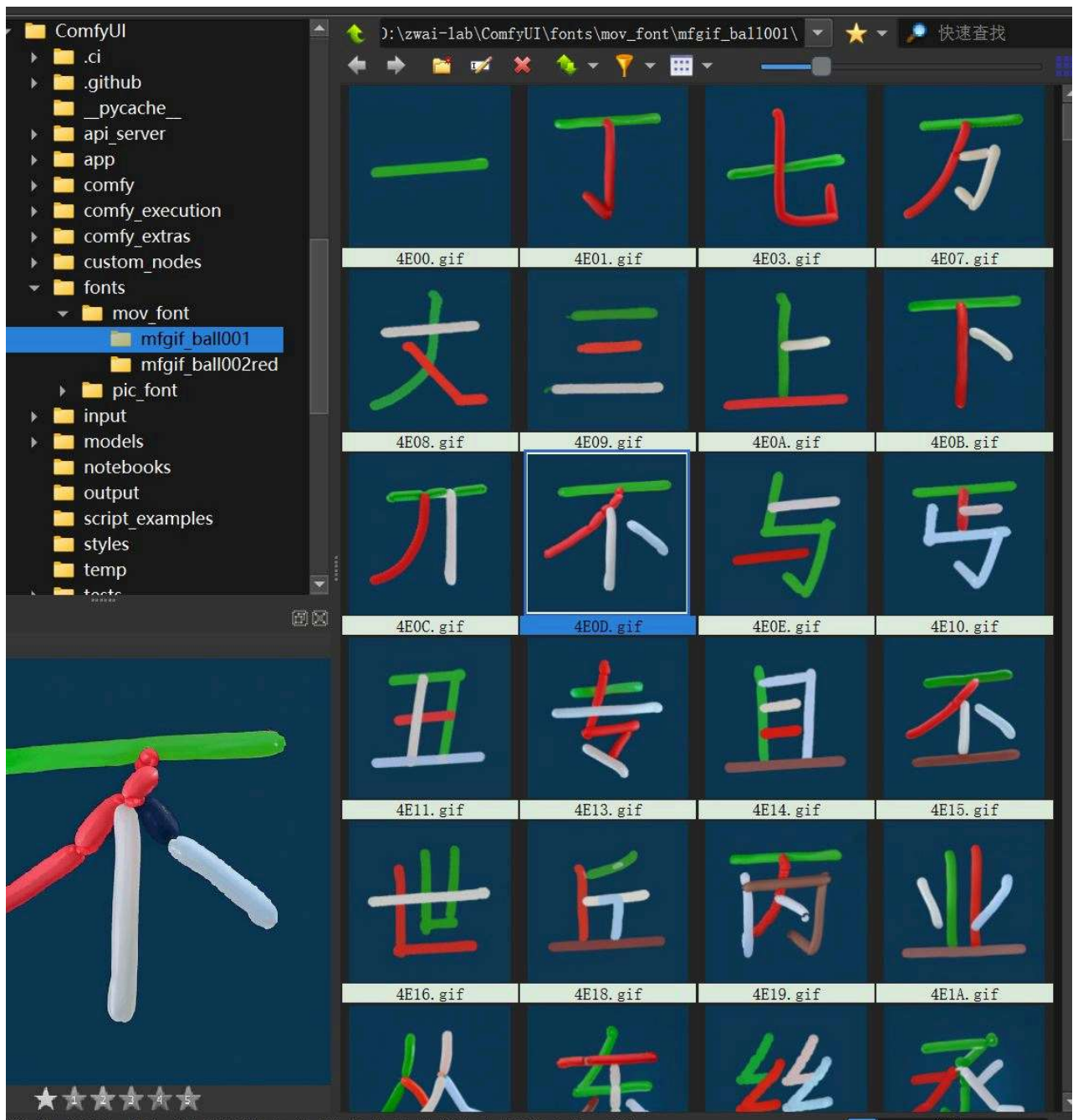
- zw-font: 本地标准ttf字体文件接口
- zw-font-pic: 本地图像字体文件接口
-
- zw-font-api: 远程服务器api接口，云字库模式
- zw-font-ucode: 内置字符串-ucod转换节点
- zw-font-mov: 隐含节点，未开放，动态视频字体接口

assets资源目录

- 有demo-workflow工作流
- 配套的图像素材
- 两套开源中文字体：字王大藏经开源字体，阿里普惠体
- 默认ttf字体文件，须copy复制到以下目录：

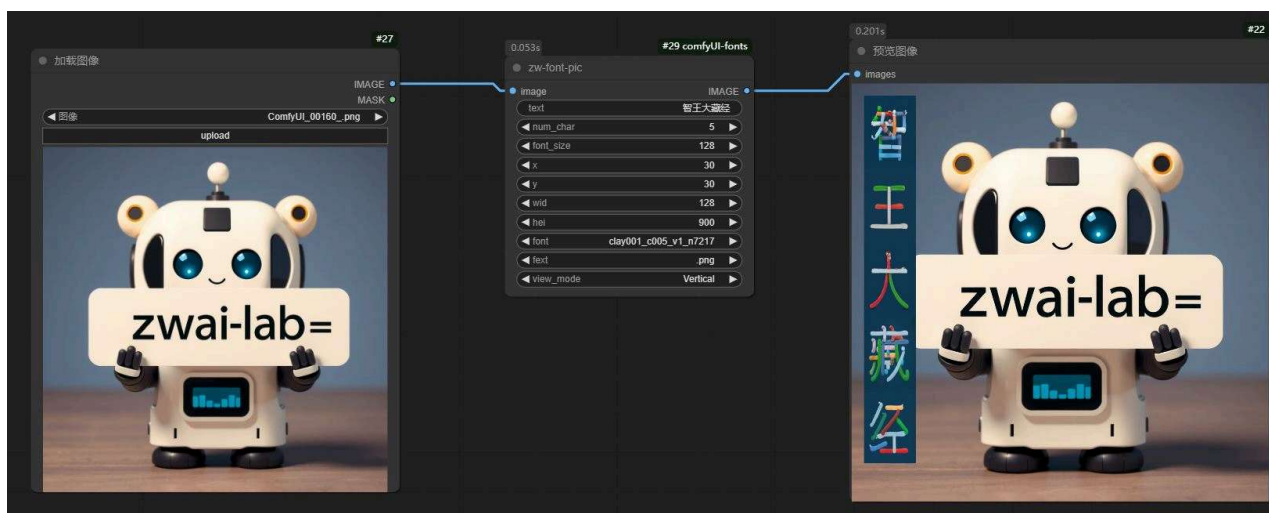
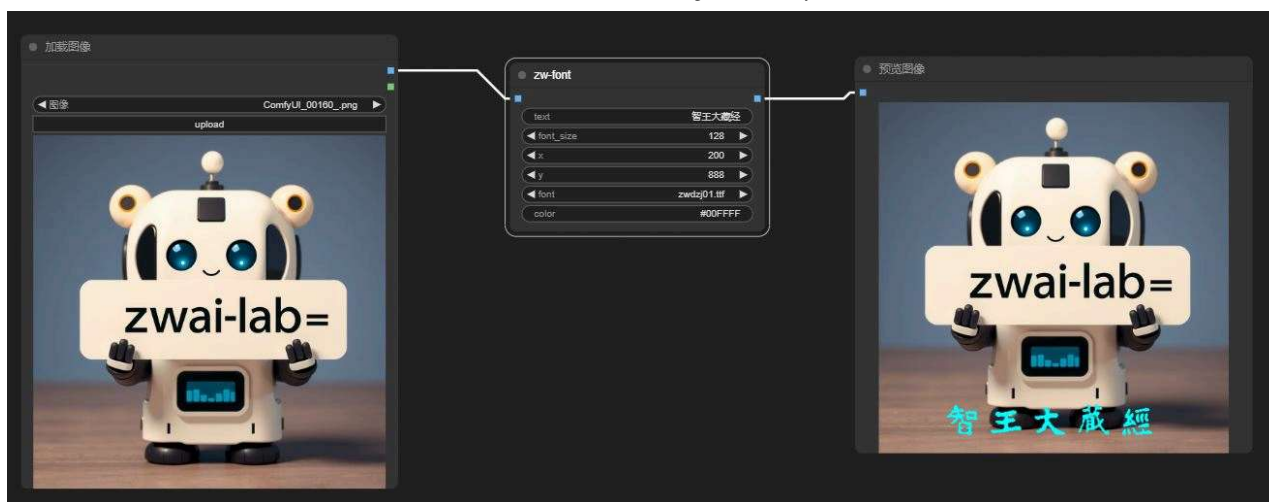
- ComfyUI/字体





主要节点-demo

参见assets资源目录，有demo-workflow workflow



```
To see the GUI go to: http://127.0.0.1:8188
FETCH DATA from: D:\zwai-lab\ComfyUI\custom_nodes\ComfyUI
got prompt
@text: '智王大藏经'
@tss: '667A,738B,5927,85CF,7ECF'
@ulst: ['667A', '738B', '5927', '85CF', '7ECF']
Prompt executed in 0.03 seconds
```

补充说明

释放

未发布版本
[创建新版本](#)

包

未发布
包 发布[您的第一个包](#)

语言

● 蟒 100.0%

建议的工作流程

基于您的技术堆栈

**皮林特**

配置

使用 pylint 对 Python 应用程序进行 lint 检查。

**Python 应用程序**

配置

创建并测试 Python 应用程序。

**使用 Anaconda 的 Python 包**

配置

使用 Anaconda 进行包管理，在多个 Python 版本上创建和测试 Python 包。

[更多工作流](#)

[关闭建议](#)