

- 环境配置
- 运行系统
- 项目架构
  - 主界面

## 环境配置

创建conda环境

```
conda create -n block_agent python=3.10 -y
```

安装依赖包

```
conda activate block_agent
pip install -r requirements.txt
```

## 运行系统

配置模型参数，打开 `.env` 文件

需要配置如下几个红框位置的参数

```
OPENAI_API_KEY=sk-... 生成解决方案用的模型的apikey
OPENAI_BASE_URL=https://xiaoai.plus/v1 接口地址
LLM_MODEL_NAME=claude-3-5-sonnet-20241022 模型名称

INDENT_API_KEY=sk-... 意图识别用的模型的apikey, 可以用差点的
INDENT_BASE_URL=https://api.deepseek.com/v1
INDENT_MODEL_NAME=deepseek-chat

VL_API_KEY=sk-... 多模态模型的apikey
VL_BASE_URL=https://xiaoai.plus/v1
VISION_MODEL_NAME=claude-3-5-sonnet-20241022
```

运行代码，在项目根路径下运行：

```
(block_agent) e:\01-personal\00-learn\01-AI\03-大模型\Agent\block_multi_agents>streamlit run main.py
```

```
streamlit run main.py
```

## 项目架构

---

- **agents** 文件夹：
  - **plan\_agent.py**: 生成拼图方案的代码
  - **vision\_agent.py**: 解析积木和目标图形的代码
- **core** 文件夹：
  - **graph.py**: langgraph图代码，将多个agent的交互和运行以图的形式构建
  - **display.py**: streamlit展示中间运行过程的组件
- **data** 文件夹: 存放上传的图片
- **utils** 文件夹：
  - **image\_processing.py**: 解析上传图片，便于发送给模型
  - **logger.py**: 日志模块

## 主界面

---

# 智能积木拼图系统

上传积木图片



Drag and drop file here

Limit 200MB per file • PNG, JPG, JPEG

Browse files



input19.png 40.8KB



上传积木框架图片



Drag and drop file here

Limit 200MB per file • PNG, JPG, JPEG

Browse files

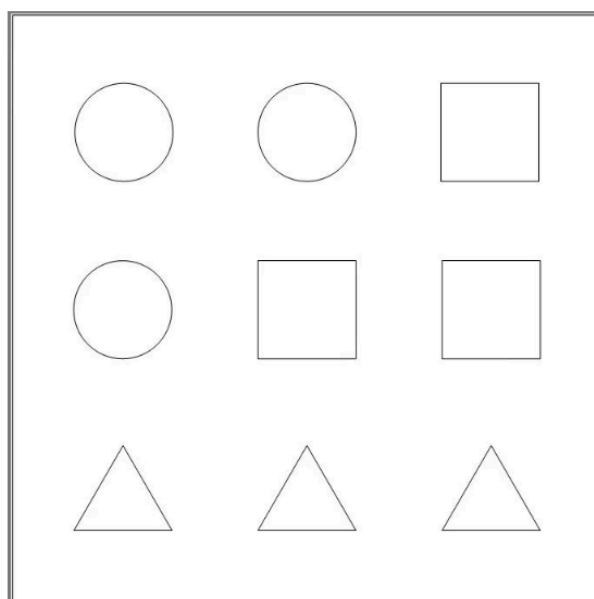
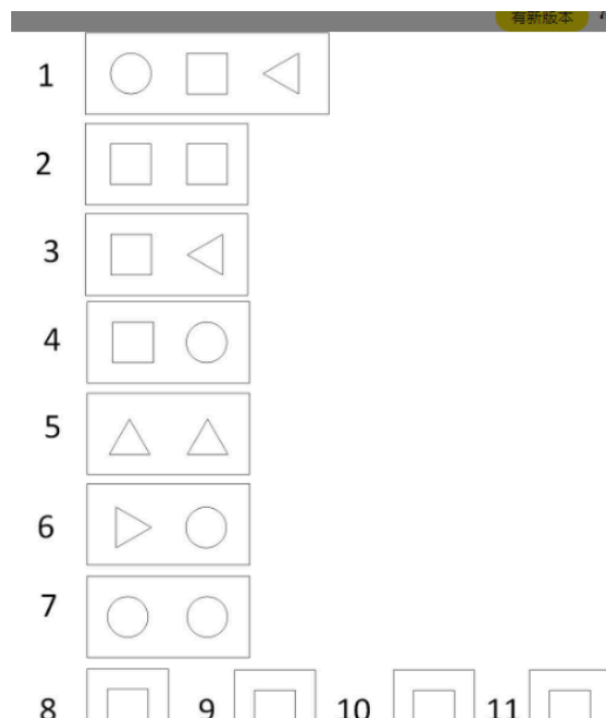


积木框架.jpg 24.7KB



任务描述

使用19块积木拼成3个正方形



上传的积木框架图片

按照提示，上传积木图片，积木框架图片和任务描述：

使用19块积木拼成3个正方形

点击边上后出现开始分析按钮，点击即可开始运行

\*\*\* CONNECTING

上传的积木图片

开始分析

开始执行多智能体工作流...

blocks\_data 详细信息

planned\_solution 详细信息

✔ workflow 执行完成!

Vision分析

任务规划

方案执行

✔ 结果验证

完整结果:

===== 方案如下: =====

【第1个目标图形】:

(7, <a b>)

(4, <d e>)

(2, <c f>)

(5, <h i>)

(12, <g>)

【第2个目标图形】:

(1, <b e h>)

(6, <d g>)

(8, <f>)

(9, <c>)

(16, <a>)

(13, <i>)

【第3个目标图形】:

(3, <f i>)

会展示中间的模型运行过程信息