大模型 agent 模式配置及应用

2026 年 10 月 12 日（第 5 周周一）布置 · 不计分，属“入场达标实验”，必须通过验收方可进入后续实验

一、全班统一动作模板（15 组同构，解读对象各组不同）

① 在 VS Code 中安装 Cline 插件并填入 API Key，切换到 agent 模式 → ② 让 AI 解读本组任务书指定的代码文件 → ③ 把 AI 的回答逐条与代码对照，标注“对 / 错 / 存疑” → ④ **至少纠正 AI 一处错误**，写明代码依据 → ⑤ 交一页“AI 解读 + 人工核对”记录（含 AI 原话与核对结论）。

二、工具与账号

VS Code + Cline 插件（cline.bot）+ **GLM 开放平台（open.bigmodel.cn）或 DeepSeek（platform.deepseek.com）任一** API Key，注册即有免费额度。**验收看命令行 agent 的真实调用证据，用网页版对话代替不算通过。**

三、证据三件套（缺一件 = 无证据 = 未完成）

命令全文（你敲的是哪条命令，原样复制）+ **完整日志**（终端从开始到结束的完整输出，不许截选）+ **产物路径**（跑出来的文件在哪、叫什么）。实验记录里只许出现你亲手跑出来的内容，禁止把 AI 的回答直接粘进去。

四、提示词四要素（各组第 3 栏的第一轮提示词即按此写就）

- **具体**：说清文件路径、问哪几个问题，别让 AI 猜；
- **给上下文**：你的系统、已装环境、报错原文；
- **要解释**：要求引用代码行号——这是你“看得懂”的保险；
- **迭代追问**：第一次不满意就换个问法再问，两次答案打架的地方就是你必须翻代码的地方。

五、三条纪律（违反任何一条，视同作业未完成）

- **不许只抄不看**：交付物会被随机抽问，讲不清楚就是没做；
- **实验数字不许抄 AI 给的**：准确率、误差、耗时必须是自己代码在自己机器上跑出来的；
- **AI 会一本正经地胡说**（幻觉）：它说的不算数，你跑出来才算数。

六、15 个小组任务索引（每人只精读自己组那一页）

组号	组名	本组实验一解读对象 / 交付物	页码
1	六目拼接 · 匹配决策组	group04_mc/train.py 三问（数据/结构/输出）	2
2	六目拼接 · 特征质量组	head_specs.py 三头规格 → 规格说明书	3
3	六目拼接 · 缝合融合组	stitch_initialize.cpp 增益补偿段 + 写死参数清单	4
4	六目拼接 · 数据与标签组	label_builder.py 三段标签 → 数据流图	5
5	六目拼接 · 测试集与评测组	ab_compare.py + run_regression.ps1 → 指标说明书	6
6	六目拼接 · 端到端数据组	datasets/README.md + make_demo_data.py → 差异说明	7
7	六目拼接 · 端到端训练组	train.py 结构与损失 → 数据流图	8
8	六目拼接 · 端到端优化组	引擎通道耗时构成 → 性能预算书	9
9	水下双目 · 数据组	video_check.py + 实跑体检 + 30 帧人工看图	10
10	水下双目 · 部署组	FoundationStereo 环境清单逐条核实	11
11	水下双目 · 训练组	train_uw_fresh.yaml → 配置说明书	12
12	水下双目 · 评测组	EPE/bad-N/D1 白皮书 + 1 对手工验算	13
13	数据基座 · 数据结构化组	pdfplumber 抽取原理 + 3 页逐格核对	14
14	数据基座 · 智能问答与图检索组	agent 调工具走通“查询-回答-引用”	15
15	数据基座 · 虚假内容识别与对抗训练组	AI 当造假者造 10 条 + 双份判定对账	16

第 1 组 · 项目一 · 六目拼接（360Stitcher）

匹配决策组

实验一

2026-10-12 布置

1 实验一的任务（一句话）

配置 GLM 编程助手的 agent 模式，让它解读 group04_mc/train.py 并回答三问，全组逐条人工核对、至少纠正一处错误。

2 本组工作目录（文件夹）

```
stitcher_cpp/training/group01_policy/
```

```
stitcher_cpp/training/group04_mc/
```

```
stitcher_cpp/training/group05_geometry/
```

三个目录各含 train.py + README.md，结构相同。目录名中的 group01/04/05 是源工程头的原始编号，与课程“第 1 组”无关。

3 第一轮提示词建议（发给 GLM，可直接复制）

▼ 复制以下整段，作为发给 GLM 的第一轮提示词

你是运行在 VS Code Cline 中的 agent 编程助手（GLM 大模型）。请打开并通读 stitcher_cpp/training/group04_mc/train.py，用中文回答三个问题：(1) 训练数据从哪来——读哪个目录、哪些文件、哪些列；(2) 网络结构长什么样——几层、每层多少神经元、什么激活函数；(3) 输出是什么——张量形状与每个数的含义。每条结论必须引用具体代码行号；拿不准就明说“存疑”，不许猜；不要背教科书式的通用答案。

4 验收标准：满足全部条件才可结束迭代

- ☐ agent 模式配置成功，有一次真实调用记录（命令行证据，网页版对话不算）
- ☐ 三问答案齐备，AI 回答原文已留档
- ☐ AI 回答逐条与代码对照，标注“对 / 错 / 存疑”
- ☐ 至少纠正 AI 一处错误，写明在代码里找到的依据（如 AI 脑补 GPU/dropout 而实际是 CPU 小网络）
- ☐ 换个问法把三问再问一遍，两次答案打架的地方已翻代码核实

5 实验一报告需要写的内容

- 一页“AI 解读 + 人工核对”记录：AI 原话摘录、三问答案
- 逐条核对结论（对/错/存疑）与纠正的错误及其代码依据
- 配置过程简记：安装 Cline、填入 GLM API Key 的命令与日志

6 产生 / 更新文件后，需要更新与上传什么

- 核对记录存入本组工作目录（如 group04_mc/ 下独立 md，或并入该目录 README.md）
- 实验一证据（配置命令、完整对话日志、产物路径三件套）归入“证据包-实验一”，学期末随总清单提交
- 本实验不产生模型交付，无需改动 stitcher_cpp/build/Release/models/，也暂不涉及第 5 组双台账登记（台账自模型交付起）

第 2 组 · 项目一 · 六目拼接（360Stitcher）

特征质量组

实验一

2026-10-12 布置

1 实验一的任务（一句话）

配置 GLM agent，逐头解读 training/common/head_specs.py 中本组三个头的规格，人工核对后汇成一页“规格说明书”。

2 本组工作目录（文件夹）

> `stitcher_cpp/training/`（本组三头子目录：`feature_uncertainty / descriptor_softening / temporal_state`）

> `stitcher_cpp/training/common/head_specs.py`（三头规格的单一事实来源）

规格说明书是你们组后面三个多月随查随用的字典，每个结论都要能指回 head_specs.py。

3 第一轮提示词建议（发给 GLM，可直接复制）

▼ 复制以下整段，作为发给 GLM 的第一轮提示词

你是运行在 VS Code Cline 中的 GLM agent。请通读 stitcher_cpp/training/common/head_specs.py，只针对 feature_uncertainty、descriptor_softening、temporal_state 三个头，逐头回答：(1) 输入张量每一维是什么（如 [B,T,24] 的 B、T、24 各指什么，T 的滑窗从哪来，不要说成“固定值”）；(2) 输出的每个数是什么含义（注意 [B,4]×2 是两个输出，不是八维向量）；(3) 用什么损失函数。每条注明代码行号；张量形状必须逐字对照代码，AI 念错形状是常见幻觉。

4 验收标准：满足全部条件才可结束迭代

- ☐ 三个头 × 三问全部答完，AI 原话留档
- ☐ 逐条与 head_specs.py 对照，标注“对 / 错 / 存疑”，至少纠正一处错误
- ☐ 规格说明书成文一页：输入尺寸 / 输出含义 / 损失函数，全部以 head_specs.py 为准
- ☐ 说明书里每句话都能在代码里指出出处（不许把 AI 回答直接粘进说明书）

5 实验一报告需要写的内容

- 一页规格说明书（实验一产物，第 19 周随报告附上）
- AI 解读与人工核对记录：AI 原话、逐条结论、纠正记录

6 产生 / 更新文件后，需要更新与上传什么

- 规格说明书存本组工作目录，并作为第 19 周提交清单“规格说明书一页”的底稿
- 实验一证据三件套（命令+完整日志+产物路径）归入“证据包-实验一”

第 3 组 · 项目一 · 六目拼接（360Stitcher）

缝合融合组

实验一
2026-10-12 布置

1 实验一的任务（一句话）

配置 GLM agent，解读 src/stitch/stitch_initialize.cpp 的增益补偿段，把“写死的数”列成清单——它们就是 gain_field 头未来要接管的对象。

2 本组工作目录（文件夹）

`stitcher_cpp/training/`（本组三头子目录：`seam_uncertainty / gain_field / fusion`）

`stitcher_cpp/src/stitch/stitch_initialize.cpp`（实验一解读对象）

先看通透改造对象：为什么亮度不均、现在的补偿逻辑是什么、哪些是拍脑袋写死的数。

3 第一轮提示词建议（发给 GLM，可直接复制）

▼ 复制以下整段，作为发给 GLM 的第一轮提示词

你是运行在 VS Code Cline 中的 GLM agent。请在 stitcher_cpp/src/stitch/stitch_initialize.cpp 中定位“增益补偿”相关段落并回答三问：(1) 为什么各路画面亮度会不均（曝光、增益、朝向差异在代码中如何体现）；(2) 现在的补偿按什么逻辑计算系数；(3) 哪些常数是写死的——逐个列出变量名、数值与行号。注意：C++ 的模板与宏容易看走眼，不要把标定参数当成运行时补偿系数；每条结论引用行号，我会在文件里逐条对账。

4 验收标准：满足全部条件才可结束迭代

- □ 三问答案齐备，AI 引用的行号逐条在文件里对上才算数
- □ “写死参数清单”成表（变量名 / 数值 / 行号 / 作用）
- □ 逐条核对标注“对 / 错 / 存疑”，至少纠正一处错误（AI 读 C++ 爱脑补变量名与模板参数）
- □ 换个问法再问一遍，答案打架处已翻代码核实

5 实验一报告需要写的内容

- 一页“AI 解读 + 人工核对”记录，含行号引用、三问答案、写死参数清单

6 产生 / 更新文件后，需要更新与上传什么

- 解读记录与写死参数清单存本组工作目录（gain_field 头设计文档的素材）
- 实验一证据三件套归入“证据包-实验一”

第 4 组 · 项目一 · 六目拼接（360Stitcher）

数据与标签组

实验一
2026-10-12 布置

1 实验一的任务（一句话）

配置 GLM agent，解读 tools/label_builder.py 三段标签（match/geometry/policy）的数据来源与字段含义，画出“遥测→标签”数据流图。

2 本组工作目录（文件夹）

```
stitcher_cpp/training_data/
```

```
stitcher_cpp/tools/（label_builder.py）
```

三段标签分别喂第 1–3 组；这张数据流图是你以后每一步操作的底图。

3 第一轮提示词建议（发给 GLM，可直接复制）

▼ 复制以下整段，作为发给 GLM 的第一轮提示词

你是运行在 VS Code Cline 中的 GLM agent。请通读 stitcher_cpp/tools/label_builder.py，分别对 match、geometry、policy 三段标签回答：(1) 这段标签从哪股遥测数据来（哪个目录、哪些文件、哪些列）；(2) 输出的每个字段是什么含义——三段有同名不同义的字段，请分开讲清楚；(3) 校验逻辑查什么、查不过怎么办。每条引用代码行号；字段含义不要背教科书，逐字对照代码。

4 验收标准：满足全部条件才可结束迭代

- □ 三段标签逐字段与代码对照核对，标注“对 / 错 / 存疑”，至少纠正一处（AI 对字段含义张口就来）
- □ “遥测→标签”数据流图完成，且校验环节画在图上（校验是生产线的质检站，漏画不算完成）
- □ AI 原话与三问答案留档

5 实验一报告需要写的内容

- 一页“AI 解读 + 人工核对”记录
- 一张“遥测→标签”数据流图（工具不限，手绘拍照也行）

6 产生 / 更新文件后，需要更新与上传什么

- 数据流图与脚本说明存 training_data/（第 19 周清单“数据流图与脚本说明”即此产物）
- 实验一证据三件套归入“证据包-实验一”

第 5 组 · 项目一 · 六目拼接（360Stitcher）

测试集与评测组

实验一
2026-10-12 布置

1 实验一的任务（一句话）

配置 GLM agent，解读 tools/ab_compare.py 与 tools/run_regression.ps1 两个评测工具，写出全组第一份武器：一页“评测指标说明书”。

2 本组工作目录（文件夹）

`stitcher_cpp/tools/`（`ab_compare.py`、`run_regression.ps1`）

代码基座两条路线都在本机：`stitcher_cpp`（九头路线）与 `e2e_gpu`（端到端路线）

A/B 统一用本组冻结的测试集；说明书是全项目指标口径的起点。

3 第一轮提示词建议（发给 GLM，可直接复制）

▼ 复制以下整段，作为发给 GLM 的第一轮提示词

你是运行在 VS Code Cline 中的 GLM agent。请分别解读：(A) stitcher_cpp/tools/ab_compare.py——它读哪两份文件？逐项列出它对比的指标（初始化耗时、稳态均值、稳态 P95、输出宽高等），每个指标标注“越高越好还是越低越好”，并解释为什么强调“一次只换一个模型”；(B) tools/run_regression.ps1——回归链有哪几环（配置、编译、单元测试、基准拼接、遥测），每一环失败会发生什么。所有指标逐项对照代码里的指标列表报行号，不要张冠李戴。

4 验收标准：满足全部条件才可结束迭代

- □ 两个工具的拆解齐备：ab_compare.py 三问 + 回归链环节及失败后果
- □ 每项指标写清中文名、方向（越大越好/越小越好）、进台账的格式
- □ 逐条与代码对照核对，至少纠正一处（常见坑：把初始化耗时混进每帧耗时——加载只发生一次）
- □ 评测指标说明书成文一页

5 实验一报告需要写的内容

- 一页评测指标说明书 + “AI 解读 + 人工核对”记录（AI 原话、逐条结论、纠正记录）

6 产生 / 更新文件后，需要更新与上传什么

- 说明书存 stitcher_cpp/tools/ 旁的本组工作区，其中指标名与方向即双台账列头，后续登记沿用
- 实验一证据三件套归入“证据包-实验一”
- 本组学期核心职责是维护双台账（数据量台账 + 基线指标台账）——实验一先定口径，台账自各组首次交付起记

第 6 组 · 项目一 · 六目拼接（360Stitcher）

端到端数据组

实验一
2026-10-12 布置

1 实验一的任务（一句话）

配置 GLM agent，解读端到端路线训练数据需求（datasets/README.md 与 make_demo_data.py），与九头路线对照，写一页“两路线数据差异说明”。

2 本组工作目录（文件夹）

`e2e_gpu/training/group10_e2e_data/`

`e2e_gpu/training/datasets/`（数据装载位，目录格式以 `datasets/README.md` 为准）

group10 是源工程原始编号，与课程“第 6 组”无关；读写图一律走 io_cn.py 的 imread_cn/imwrite_cn（中文路径防静默失败）。

3 第一轮提示词建议（发给 GLM，可直接复制）

▼ 复制以下整段，作为发给 GLM 的第一轮提示词

你是运行在 VS Code Cline 中的 GLM agent。请阅读 e2e_gpu/training/datasets/README.md 与 make_demo_data.py，回答三问：(1) 一个训练样本长什么样（left/right 同名成对、512×512；口径是“对内前一路 / 后一路”，不是左相机右相机）；(2) 标签是什么（合成对的已知平移真值、真实对的伪真值变换——“端到端不需要标签”是错误说法）；(3) 训练与验证怎么划分。然后与第 4 组的遥测标签对照，按“样本形态 / 标签来源 / 数据量单位 / 质量瓶颈”四行列成对照表。

4 验收标准：满足全部条件才可结束迭代

- □ 三问答案 + 四行对照表齐备，每个格子落到具体文件与字段（“数据不一样”级别的空话不算完成）
- □ 逐条与文件核对，标注“对 / 错 / 存疑”，至少纠正一处错误
- □ AI 原话留档，差异说明成文一页

5 实验一报告需要写的内容

- 一页“两路线数据差异说明” + AI 解读与人工核对记录

6 产生 / 更新文件后，需要更新与上传什么

- 差异说明存 group10_e2e_data/（第 19 周数据集卡片的底稿）
- 实验一证据三件套归入“证据包-实验一”
- 如试跑 make_demo_data.py 产生新数据批次：按 datasets/ 目录规范建夹，批次名标注“合成”

第 7 组 · 项目一 · 六目拼接（360Stitcher）

端到端训练组

实验一
2026-10-12 布置

1 实验一的任务（一句话）

配置 GLM agent，解读 training/train.py 的网络结构与损失函数（重投影误差），画出“图像对→变换估计”数据流图。

2 本组工作目录（文件夹）

> `e2e_gpu/training/`（本组任务夹 `group11_e2e_train`：`train.py`、`export_onnx.py`、`deliver_model.bat` 在 `training/` 根目录）

group11 是源工程原始编号；形状契约：输入 [B,2,3,512,512]、输出 [B,3,3]、opset 13。先知道要训什么，再谈怎么训。

3 第一轮提示词建议（发给 GLM，可直接复制）

▼ 复制以下整段，作为发给 GLM 的第一轮提示词

你是运行在 VS Code Cline 中的 GLM agent。请通读 e2e_gpu/training/train.py，回答三问：(1) 网络骨架长什么样——几层、卷积核多大、残差连接在哪（注意：输出是 3×3 变换矩阵的 9 个数，不是分类得分）；(2) 输入输出张量形状及每一维的含义；(3) 损失函数怎么定义——重投影误差如何逐像素计算。然后画出“图像对→变换估计”数据流：左右图 → 骨干提特征 → 融合 → 回归 3×3 矩阵 → 重投影误差，预处理（缩放 512、归一化）必须画上。每条结论引用行号，以代码为准。

4 验收标准：满足全部条件才可结束迭代

- □ 三问答案齐备且引用行号，逐条核对标注“对 / 错 / 存疑”
- □ 数据流图完成且包含预处理环节（预处理不对，后面全错）
- □ 至少纠正 AI 一处错误（如把它讲成分类网络、损失函数名张口就来）

5 实验一报告需要写的内容

- 一页“AI 解读 + 人工核对”记录 + 一张数据流图（手绘拍照亦可）

6 产生 / 更新文件后，需要更新与上传什么

- 数据流图存 group11_e2e_train/——第 19 周清单“数据流图与训练记录”即此产物
- 实验一证据三件套归入“证据包-实验一”
- 4080 机器只归本组与导师使用：实验一无训练任务不需预约；今后用时先登记

第 8 组 · 项目一 · 六目拼接（360Stitcher）

端到端优化组

实验一
2026-10-12 布置

1 实验一的任务（一句话）

配置 GLM agent，解读引擎一次调用的耗时构成（预处理 / 骨干前向 / 后处理），写一页给 800ms 立规矩的“性能预算书”。

2 本组工作目录（文件夹）

`e2e_gpu/training/group12_e2e_optim/`

`e2e_gpu/cpp/src/`（引擎通道源码，实验一解读对象）

group12 是源工程原始编号，与课程“第 8 组”无关。引擎侧先验只当假设，一切以本机实测为准。

3 第一轮提示词建议（发给 GLM，可直接复制）

▼ 复制以下整段，作为发给 GLM 的第一轮提示词

你是运行在 VS Code Cline 中的 GLM agent。请阅读 e2e_gpu/cpp/src/ 中端到端引擎通道的源码，回答三问：(1) 一次调用前后各做了什么——预处理（缩放 512、归一化等）与后处理（5 对变换链式累积与渲染）；(2) 骨干网络前向在算什么；(3) 你估计预处理 / 前向 / 后处理三段各占多少耗时，依据是什么（不许拿通用模型的经验数字套本项目，引擎是我们自己的）。每条注明文件与行号，我会在代码里逐条对账。

4 验收标准：满足全部条件才可结束迭代

- □ 三段拆解与先验占比估计齐备，且每项占比说明了依据
- □ 性能预算书成表：800ms 的分配（预处理 / 引擎≤300ms / 渲染 / 显示余量），每项配一列“怎么测”（不可测量的预算只是愿望清单）
- □ 逐条核对标注“对 / 错 / 存疑”，至少纠正一处（常见坑：AI 把“模型推理”当唯一耗时，漏掉预处理与后处理）

5 实验一报告需要写的内容

- 一页性能预算书 + “AI 解读 + 人工核对”记录

6 产生 / 更新文件后，需要更新与上传什么

- 预算书存 group12_e2e_optim/——第 19 周与离场实测复盘对照提交（差异要有解释）
- 实验一证据三件套归入“证据包-实验一”

第 9 组 · 项目二 · 水下双目自动标记（HYSY123 FPSO）

数据组

实验一
2026-10-12 布置

1 实验一的任务（一句话）

配置 GLM 类命令行 agent，实跑“视频体检脚本”产出体检报告，并亲自看 30 帧核对抽帧排序是否合理。

2 本组工作目录（文件夹）

`tasks/u09_data/`（本组任务夹，一切产出放这里）

主线 `code/` 只读不改（解读对象 `code/tools/video_check.py`）

真实数据（500 对锚链校正立体图对开局资产包）在 E:\锚链视频\stereo_dataset；数据集与划分共享给第 10/11/12 组。

3 第一轮提示词建议（发给 GLM，可直接复制）

▼ 复制以下整段，作为发给 GLM 的第一轮提示词

你是运行在 VS Code Cline 中的 GLM agent。请解读主线 code/tools/video_check.py 的参数：(1) 场景切割阈值怎么设、依据什么；(2) 模糊度与低照度分别怎么判；(3) 输出的“最值得抽帧”排序按什么规则。每条引用行号。然后帮我在 tasks/u09_data/ 下写运行配置，对一段样例视频实跑：python code/tools/video_check.py --input demo.mp4 --out tasks/u09_data/check/。注意：你没有看过这段视频，体检数字必须来自我的终端输出，不许替我“想象”结果。

4 验收标准：满足全部条件才可结束迭代

- □ agent 解读逐条与源码对照，标注“对 / 错 / 存疑”，至少纠正一处
- □ 体检脚本实跑成功（先解决 ffmpeg：命令行 ffmpeg -version 必须能出版本号，PATH 没配就加上再重开终端）
- □ 按“最值得抽帧”排序的体检报告落地
- □ 30 帧人工看图完成：糊的、黑的确实排在后面——AI 不可代看，这半小时没有人能替你

5 实验一报告需要写的内容

- agent 解读 + 人工核对记录一页（AI 原话、核对结论、至少一处纠正）
- 体检报告（场景切割 + 模糊度/低照度统计 + 抽帧优先级排序）
- 30 帧看图核对记录

6 产生 / 更新文件后，需要更新与上传什么

- 运行配置与体检报告存 tasks/u09_data/check/
- 实验一证据三件套归入“证据包-实验一”（学期末四包之一）
- 涉密红线：原始视频/帧不上云盘、不进在线 AI——体检统计数字可以进提示词，视频与帧本体绝对不行

第 10 组 · 项目二 · 水下双目自动标记（HYSY123 FPSO）

部署组

实验一
2026-10-12 布置

1 实验一的任务（一句话）

让 agent 生成“4080 环境配置清单”草稿，逐条对照 FoundationStereo 官方仓库核实，定稿存 tasks/u10_deploy/env_checklist.md。

2 本组工作目录（文件夹）

`tasks/u10_deploy/`（本组任务夹；清单定稿位置 `env_checklist.md`）

主线 `code/` 只读不改

清单是整个学期的施工图——错一条，实验二还你一夜；本组另兼 4080 时段台账（全队预约都记在这里）。

3 第一轮提示词建议（发给 GLM，可直接复制）

▼ 复制以下整段，作为发给 GLM 的第一轮提示词

你是运行在 VS Code Cline 中的 GLM agent。请阅读 FoundationStereo 官方仓库的 README、环境文件与 requirements.txt，生成 4080 环境配置清单草稿：Python 版本、PyTorch 与 CUDA 版本、flash-attn 版本、其余依赖。要求：(1) 每条注明出处（仓库哪个文件哪一段）；(2) 每个版本组合说明是“官方明确要求”还是“你的常见搭配推测”——编造不存在的版本组合是典型幻觉；(3) 每条后面补一行“怎么验证装好了”，如 python -c "import torch; print(torch.cuda.is_available())" 应输出 True。

4 验收标准：满足全部条件才可结束迭代

- □ 清单草稿完成（AI 产出）
- □ 逐条到官方仓库核对并标注“对 / 错 / 存疑”——以仓库为准，不以 AI 为准
- □ 每条补上验证命令，核对日期写进清单
- □ 终版清单定稿存 tasks/u10_deploy/env_checklist.md（三样齐全：草稿、核对标记版、带验证命令的终版）

5 实验一报告需要写的内容

- 三样清单的归档说明 + 逐条核对记录（AI 编造或过时的条目是如何被发现的）

6 产生 / 更新文件后，需要更新与上传什么

- 终版清单定稿存 tasks/u10_deploy/env_checklist.md——实验二部署直接按它施工
- 实验一证据三件套归入“证据包-实验一”

第 11 组 · 项目二 · 水下双目自动标记（HYSY123 FPSO）

训练组

实验一
2026-10-12 布置

1 实验一的任务（一句话）

配置 GLM agent，逐项解释 configs/train_uw_fresh.yaml（lr/bs/iters/epoch），人工核对后写成 1–2 页“配置说明书”。

2 本组工作目录（文件夹）

`tasks/u11_train/`（本组任务夹；配置先拷贝到这里，改也只改副本）

主线 `code/` 与 `configs/` 只读

战场纪律：先基线后微调；训练闸门（epoch 数、启动时机）须经小助手批准。

3 第一轮提示词建议（发给 GLM，可直接复制）

▼ 复制以下整段，作为发给 GLM 的第一轮提示词

你是运行在 VS Code Cline 中的 GLM agent。我已把 configs/train_uw_fresh.yaml 拷贝到 tasks/u11_train/，请逐项解释每个键：控制什么、默认值多少、往大调 / 往小调各会怎样。注意：(1) iters 与 epoch 不是一回事——一个数“总共走多少步”，一个数“全量数据走几遍”，请给出两者的换算关系；(2) 每项解释你要在主线代码里找到读取该键的那一行并报行号，找不到就明说“存疑”；(3) 最后每项配一句人话，如 lr 是“步子迈多大”、bs 是“一口吃几对图”。

4 验收标准：满足全部条件才可结束迭代

- □ 逐项解释完成，且每项在主线代码里找到读取行、行号写进说明书（只解释不核对不算通过）
- □ iters 与 epoch 的关系已在说明书里算清
- □ 每项有“往大往小会怎样”的判断与一句人话翻译（抄成名词字典不行）
- □ 逐项核对标注“对 / 错 / 存疑”，至少纠正一处，说明书成文 1–2 页

5 实验一报告需要写的内容

- 配置说明书（第 19 周随报告定稿提交）+ 逐项核对记录（含代码行号与对/错/存疑）

6 产生 / 更新文件后，需要更新与上传什么

- 配置副本与说明书存 tasks/u11_train/
- 实验一证据三件套归入“证据包-实验一”

第 12 组 · 项目二 · 水下双目自动标记（HYSY123 FPSO）

评测组

实验一
2026-10-12 布置

1 实验一的任务（一句话）

让 agent 写“指标白皮书”（EPE / bad-1/2/4px / D1），对 10 对样例实跑指标，并手工验算其中 1 对与脚本对账。

2 本组工作目录（文件夹）

`tasks/u12_eval/`（本组任务夹；实验二的一键评测脚本也放这里）

主线 `code/` 只读不改

先把尺子造准——尺子不准，后面全错；基线数字表是全队成绩单。

3 第一轮提示词建议（发给 GLM，可直接复制）

▼ 复制以下整段，作为发给 GLM 的第一轮提示词

你是运行在 VS Code Cline 中的 GLM agent。请逐个立体匹配评测指标写清定义、公式、取值范围、适用场景：EPE（end-point error）、bad-1px/2px/4px、D1。三个要求：(1) EPE 是平均视差绝对误差，越小越好，方向不要写反；(2) bad-N 与 D1 的分母各统计哪些像素，公式里每个下标、每个绝对值、每个平均范围都要写明；(3) 找到主线代码中实现这些指标的文件与行号，逐条对照代码核对你的公式——不许照背教科书或另一种变体定义。

4 验收标准：满足全部条件才可结束迭代

- □ 白皮书每条定义都有出处（主线代码 / 核对过的文献）
- □ 10 对样例实跑指标脚本，全部输出保存
- □ 1 对手工验算完成：把视差真值图缩成 10×10，用计算器把 100 个像素的绝对误差加起来除以 100，与脚本算出的 EPE 对上账（手工验算不可跳过——脚本错一个绝对值，往后所有数字都是错的）
- □ AI 公式的下标与符号幻觉已逐条对账纠正（至少一处）

5 实验一报告需要写的内容

- 指标白皮书（含每条定义的出处）
- 10 对样例的指标输出
- 1 对手工验算过程（可拍照存档）

6 产生 / 更新文件后，需要更新与上传什么

- 白皮书与验算记录存 tasks/u12_eval/——白皮书第 19 周随清单提交
- 实验一证据三件套归入“证据包-实验一”

第 13 组 · 项目三 · 数据基座治理（海洋石油平台）

数据结构化组

实验一
2026-10-12 布置

1 实验一的任务（一句话）

配置 GLM agent，讲清 pdfplumber 表格抽取原理，对 3 份样例 PDF 各抽一页并逐格人工核对。

2 本组工作目录（文件夹）

> `tasks/d01_struct/`（样例 PDF 放 `data/`，产物放 `out/`，脚本放 `src/`）

上游是项目提供的样例包（3–5 份含表格与公式的工程文档 PDF 片段）；结构化产物是第 14 组问数 demo 的口粮。

3 第一轮提示词建议（发给 GLM，可直接复制）

▼ 复制以下整段，作为发给 GLM 的第一轮提示词

你是运行在 VS Code Cline 中的 GLM agent。请回答 pdfplumber 表格抽取三问：(1) extract_tables 靠什么把一张表从页面里“认”出来——线框还是文字对齐？(2) 跨页表格为什么会被拆开？(3) 什么情况下会抽错？回答必须对照 pdfplumber 源码或官方文档，注明依据，不许背教科书。然后帮我写一个十行小脚本：用 pdfplumber 对 data/ 下 3 份样例各抽一页的表格，结果存 JSON 到 out/。动手前先检查样例是文字版 PDF——扫描件没有文字层，extract_tables 会返回空列表。

4 验收标准：满足全部条件才可结束迭代

- ☐ pip install pdfplumber 装好并验证版本（命令与输出留档）
- ☐ 三问回答与源码/官方文档逐条对照，标注“对 / 错 / 存疑”，至少纠正一处（agent 会脑补原理）
- ☐ 3 页抽取结果 JSON 落盘，逐格打开 PDF 对照，标注“对 / 错 / 漏”（列名错了整张表都算错，只数行数不看列名不行）

5 实验一报告需要写的内容

- 一页“AI 解读 + 人工核对”记录（AI 原话、核对结论、至少一处纠正）
- 3 页抽取结果与逐格核对记录

6 产生 / 更新文件后，需要更新与上传什么

- 脚本存 src/、抽取 JSON 存 out/（第 19 周结构化产物与锚点表的起步素材）
- 实验一证据三件套归入“证据包-实验一”

第 14 组 · 项目三 · 数据基座治理（海洋石油平台）

智能问答与图检索组

实验一

2026-10-12 布置

1 实验一的任务（一句话）

配置 GLM agent，让它调用你们写好的工具函数，走通一次“查询-回答-引用”完整循环并归档——亲身体验 agent 循环。

2 本组工作目录（文件夹）

`tasks/d02_agent_kg/`（`demo/` 放问数服务，`kg/` 放图谱实验，`tests/` 放 30 题）

中期前可用项目提供的演示数据兜底（fpso_navigation 等四个查询模型的精简 CSV）。

3 第一轮提示词建议（发给 GLM，可直接复制）

▼ 复制以下整段，作为发给 GLM 的第一轮提示词

你是运行在 VS Code Cline 中的 GLM agent。我要亲身体验 agent 循环（模型提要求 → 程序干活 → 结果回填）。请先帮我起草三个查询工具函数：每个函数必须有清晰 docstring（每个参数什么含义、返回什么 JSON——写不清 agent 就不会用）。然后我们手写一个最简循环：你输出工具名与参数，程序执行，结果回填对话。规则：必须调用工具函数作答，禁止凭记忆背答案；每轮历史对话回填，不许失忆。第一个问题： “2023 年 11 月 18 日 20:27 的轴向振动是多少？ ” 要求给出答案 + 所用 SQL + 数据来源行号三件套。

4 验收标准：满足全部条件才可结束迭代

- □ 三个工具函数草稿完成，人核对过签名与 docstring（AI 起草、人核对）
- □ 一次完整“查询-回答-引用”循环跑通：agent 真的调了工具（不是背答案），回答带 SQL 与来源行号
- □ 完整循环记录归档（日志或录屏，含工具调用参数与返回原文）

5 实验一报告需要写的内容

- 完整循环日志 / 录屏
- AI 起草与人核对的分工说明（哪个函数 AI 写的、你怎么核对的 docstring）

6 产生 / 更新文件后，需要更新与上传什么

- 工具函数与最简循环脚本存 demo/
- 循环日志归档进“证据包-实验一”
- 安全红线：API key 绝不写进代码与提交包（交包前全文搜一遍）

第 15 组 · 项目三 · 数据基座治理（海洋石油平台）

虚假内容识别与对抗训练组

实验一

2026-10-12 布置

1 实验一的任务（一句话）

配置 GLM agent 并让它角色扮演“造假者”生成 10 条假设备振动记录（假数值由程序给定），人工与规则各判一遍并对账失误。

2 本组工作目录（文件夹）

`tasks/d03_adversarial/`（`fake/` 放造假产物，`dataset/` 放混装数据集，`model/` 放分类器）

铁律：大模型只能改说法，永远改不了事实——目标假数值由程序给定；造假配额制，一轮一记账。

3 第一轮提示词建议（发给 GLM，可直接复制）

▼ 复制以下整段，作为发给 GLM 的第一轮提示词

你是运行在 VS Code Cline 中的 GLM agent，现在扮演“造假者”。规则铁律：我先用程序把记录里某个数值乘一个系数生成假数值——你只能改“说法”（话术包装），永远不许自己发明或修改数值。请把带假数值的记录包装得以假乱真：合理的单位、口径、时间戳与解释性文字，像一条真的运维记录。第一轮：对 10 条程序给定假数值的记录各写一段包装话术，口径要“以假乱真”，不许一望便知。

4 验收标准：满足全部条件才可结束迭代

- □ 造假脚本完成：假数值确由程序给定（AI 越权自创数值的样本一律作废，不得入集）
- □ 10 条假记录生成，混入 5 条真记录，洗牌编号（不洗牌、顺序可猜，判定记录作废）
- □ 组内每人在不看答案的情况下逐条盲判并记录把握程度（先判完再对答案）
- □ 三条规则（数值越界 / 单位错 / 时间戳矛盾）由程序判完一遍
- □ 对账完成：人工错几条、规则错几条、分别错在哪，写成失误清单

5 实验一报告需要写的内容

- 15 条样本（洗牌编号）
- 人工与规则两份判定记录
- 失误清单（人工 vs 规则各自的错题与原因）

6 产生 / 更新文件后，需要更新与上传什么

- 造假脚本与样本存 fake/（后续 ≥500 条对抗数据集 v1 的种子）
- 实验一证据三件套归入“证据包-实验一”，并开始记 API 调用与花费台账（配额制：每轮造多少条要记账，超配额整轮作废）