

9 月 8 日-9 月 21 日整体计划

一、关键里程碑

日期	里程碑 / 交付物
09/08 (周一)	• 分配剩余 6 个分项打分调优任务 (给小高) • 将每个分项的新代码融入打分系统 (小金) • 同时调优直到相关性 0.8 (小高负责还没处理的四项、小金负责之前小梁弄的两项)
09/09 (周二)	• 先完整跑通所有打分流程, 要保证周三能拿到 400 张 人工美观化 前后的机器打分以及人工打分 (如果标注好了) • 在等待结果时接入 2D 数据输入管线 (小金)
09/10-11 (周三-四)	• 产出 12×12 分项相关性矩阵 + 聚类热图 (后面有说明) • 与人工分析交叉验证, 讨论聚类结果 → 定义「分项依赖 / 互斥」关系 • 输出《美观化算法分项分组》 • 基于分组进行优化 & Owner 分配
09/11 (周四)	• 各 Owner 开始进行算法参数化 & 单元测试 • 选取 150 张代表样本集进行 小规模 (150 张) 全流程跑分 + 美观化 • 收集运行时、打分变化、视觉对比
09/12 (周五)	• 复盘小规模结果, 如有需要更新下一步规划 • 继续调参 & Bugfix
09/09-11 (周二-四晚)	• 中规模 (400 张) 测试
09/12-12 (周五-日晚)	• 大规模 (2000-4000 张) 测试
09/15-19 (周一-五)	• 继续调参 & Bugfix • 当天内使用小规模测试 (150 张) • 每天晚进行中规模 (400 张) 测试
09/19 (周五)	• 整理汇报材料、准备演示 DEMO
09/19-21 (周五-日晚)	• 大规模 (2000-4000 张) 测试
09/22 (周一)	最终汇报

二、分项相关性矩阵 + 聚类示例代码

口径说明: 此矩阵基于“同一张图在**人工美观化**前后机器打分的差值 $\Delta = \text{score_after} - \text{score_before}$ ”。• 目前自动美观化尚在调优阶段, 机器美观结果不具参考价值, 因此先用人工美观版本做 Δ 。• 直观意义: $\Delta > 0$ 表示该分项因美观化得分提升, $\Delta < 0$ 表示下降。• 计算 Δ 的相关系数, 可以发现「某分项提分时是否伴随另一分项掉分」, 用于分析算法间正/负耦合。

数据结构 (必须对齐)

• 设样本量 = N (N 150~400)。读取两张 CSV 后按行对齐并计算:

- $\Delta_df = \text{post_df} - \text{pre_df} \rightarrow \text{shape} = (N, 12)$

• **行 (index):** 台区编号 / 图片 ID / 样本 ID。

• **列 (columns):** 12 个分项编号或中文名 (如 01_ 杆塔, ...12_ 台区整体混乱)。

• 相关性矩阵 = $\Delta_df.\text{corr}()$ $\text{shape} = (12, 12)$, 行列均为分项, 值是 Pearson r。

```

"""Compute correlation matrix of score deltas (after - before) and draw clustered heatmap."""
import time, pathlib, pandas as pd, seaborn as sns, matplotlib.pyplot as plt
from sklearn.preprocessing import StandardScaler

start = time.perf_counter()
# 1) 读入机器打分 (治理前 & 治理后), 两份 CSV 必须同列同行顺序
pre_path = pathlib.Path("data/machine_scores_before.csv")
post_path = pathlib.Path("data/machine_scores_after.csv")
pre = pd.read_csv(pre_path, encoding="utf-8") # shape: (N, 12)
post = pd.read_csv(post_path, encoding="utf-8")
assert pre.shape == post.shape, "Pre/Post score tables must align"

# 2) 计算差值  $\Delta$ 
delta = post - pre

# 3) 相关系数矩阵
corr = delta.corr(method="pearson")

# 4) 绘图
sns.set_theme(style="white")

g = sns.clustermap(
    corr,
    cmap="vlag",
    linewidths=0.8,
    figsize=(8, 8),
    square=True,
    cbar_kws={"label": "Pearson r ( $\Delta$  scores)"},
)
plt.title(" $\Delta$  Score Correlation & Cluster (After - Before)", pad=80)
g.figure.tight_layout()
output_dir = pathlib.Path("results/plots/20250908")
output_dir.mkdir(parents=True, exist_ok=True)
fig_path = output_dir / "delta_corr_clustermap.png"
g.savefig(fig_path, dpi=200)

# 5) 保存矩阵
corr.to_csv(output_dir / "delta_corr_matrix.csv", encoding="utf-8", index=True)
print(f"[INFO]  $\Delta$ -correlation saved to {fig_path}. Time cost: {time.perf_counter() - start:.1f}s")

```

三、资源预估

项目	数据量	耗时 (估计)	备注
小规模实验	150 张 \times 1 min	2.5 h	当天可跑完
中规模实验	400 张 \times 1 min	6.7 h	可夜间离线跑
全量 2k-4k	2.8 d-5.6 d		仅在最终或离线阶段考虑

四、风险与缓冲

1. 打分调优达不到 0.8: 提前在 09-09 晚评估, 必要时简化目标或调整权重。
2. 算法互相影响过强: 深层分析、拆解算法、逐层实验。
3. 算法调参困难: 去 @ 居博士, 讨论问题、改代码、一起找解决方案。
4. 实验运行时间长: 随时切换到小规模实验或使用 wandb 进行多机并行。

更新时间: 2025-09-08