目录

1. 项目背景和意义

1.1 项目背景

当前，教师与培训师制作高质量教学视频面临高门槛、高成本、耗时长的难题。市场上虽有 PPT 录制、AI 数字人等工具，但普遍存在模板僵硬、动画定制弱、无法深度解析原 PPT 逻辑等问题。我们的项目瞄准了这一空白，旨在实现对现有 PPT 的深度理解、元素级动画重构与 AI 讲稿精准同步，打造一款真正智能、定制化的自动化视频生产工具。项目定位为“基于 AI 与自动化工具链的 PPT 转动态教学视频生成系统”，属于 AI 赋能内容创作（AIGC） 与数字教育工具的交叉领域。该领域近年来发展迅速，国内外均有相关探索

1.1.1 国内市场与产品动态

① AI 演示文稿生成工具： 以 Gamma、Beautiful.AI、Tome 等为代表的工具已能通过自然语言描述自动生成结构完整、设计美观的 PPT。国内厂商如 WPS AI、讯飞智文、阿里通义等也推出了类似功能，它们侧重于从 0 到 1 的内容与形式生成。

② PPT 转视频工具： 微软 PowerPoint 本身就内置了“录制幻灯片演示”和“导出为视频”功能，允许录制旁白并生成视频。来画、万彩动画大师等工具，则允许用户为 PPT 元素添加丰富的动画和角色，制作成更具表现力的视频。B 站、抖音上的许多知识区 UP 主使用 Premiere、Final Cut Pro 等专业软件进行深度加工。

③ AI 数字人播报： 如腾讯智影、百度智能云曦灵、HeyGen 等平台，允许用户上传文案和 PPT，由 AI 数字人进行讲解并合成视频。这是与您项目最接近的竞品方向，但其模板化程度高，自定义动画能力弱。

④ AI 动画生成： 如 RunwayML、Pika Labs 等，可根据文本或图片生成动态视频片段，但目前难以精确控制多元素、长时间的解说类视频的逻辑与同步。

综上所述，国内市场存在“AI 生成 PPT”、“手动/模板化 PPT 转视频”和“AI 数字人播报”三条主要赛道。我们的项目创新性地将三者结合，并聚焦于 “对现有 PPT 的深度解析、元素级动画生成与 AI 讲稿精准同步” ，实现了从静态资产到高质量、定制化动态视频的自动化流水线，在自动化程度与视觉定制化之间找到了一个差异化定位。

1.1.2 国外市场与技术动态

① Synthesia、Pictory、InVideo： 是国外领先的 AI 视频生成平台。它们通常提供丰富的模板、AI 语音、Stock 素材库，用户输入文案即可生成营销、培训类视频。它们同样强于“生成”，弱于对现有复杂 PPT 文件的“深度解析与重构动画”。

② Microsoft Designer & Clipchamp： 微软正将 AI 深度集成至其创作工具链。Designer 可根据描述生成设计，Clipchamp 提供智能剪辑。未来不排除其将类似能力更深度地与 PowerPoint 结合。

③ OpenAI 的 Sora 等文生视频模型展现了惊人的世界模拟能力，但离可控、精确、长篇幅的教育内容生成仍有距离。

④ 学术界在 “多媒体文档自动讲解生成” 领域有长期研究，旨在从幻灯片自动生成解说词和简单的视觉强调，但大多停留在原型阶段，未形成成熟产品。

展望未来，多模态大模型（如 GPT-4V、Gemini）的发展，使得 AI 对 PPT 内容（图

文混合）的理解能力增强。未来的系统可能直接“看懂”PPT 页面，而无需依赖纯文本提取，从而生成更贴合的讲稿和视觉规划。

1.2 项目意义

在社会与教育方面，本项目的核心价值在于推动知识的高效传播与普惠共享。通过将静态的 PPT 课件自动化转换为生动易懂的动态讲解视频,能够显著降低高质量教学视频的制作门槛与时间成本。这一能力使得广大教师、培训师及知识工作者能够轻松地将自身的专业知识转化为可广泛传播的数字资源，有效打破传统教学的时空限制，为在线教育、混合式学习及终身学习体系提供强有力的资源支撑。同时，动态视觉与语音讲解相结合的形式，符合现代多媒体认知原理，能比单纯的文字或静态画面更有效地促进学习者的理解与记忆，从而整体提升知识传播的效能与学习体验。

同时在技术层面，本项目是 AIGC（人工智能生成内容）技术在实际生产场景中一次扎实的工程化集成与落地示范。它创新性地将大语言模型的内容生成能力、尖端语音合成技术以及强大的开源多媒体处理工具链进行深度融合，构建了一条从复杂文档到高质量视频的完整自动化流水线。这不仅为教育科技、企业培训与知识付费等领域提供了一种高效、可规模化的新型数字内容生产解决方案，也展示了利用现有工具链解决复杂跨模态问题的可行路径。项目的实践为类似“文档自动化讲解”的需求提供了宝贵的技术范式和实现参考，有望推动相关工具链的进一步创新与优化，并可能催生出基于自动化视频生产的新型服务模式与业态。

2. 需求分析

2.1 项目概述

本项目旨在开发教学 ppt 视频生成项目，能够将用户提供的静态 PPT 演示文档，全自动地转换为一部专业的、带有生动语音讲解和匹配内容动态动画的教学视频。目标用户是教师、培训师、企业内训师、知识内容创作者等非专业视频编辑人员。系统通过模块化流水线，依次完成内容提取、AI 脚本创作、语音合成、视觉元素动画生成及音视频合成等任务，最终输出一个可直接用于发布或教学的高质量 MP4 视频文件。

2.2 功能需求分析

1. **文档解析与原始素材提取功能：**内容解析与素材提取模块是系统的数据基础。该模块以 PPTX 格式文件为输入，核心任务是对幻灯片进行多维度解析：精确提取每一页的文字内容，为后续生成讲解脚本提供文本依据；将每一页幻灯片转换为高保真的静态图片，作为视频的视觉基底；同时深度分析页面结构，识别并分离出独立的图形、图像等可视化元素，并将其类型、位置与尺寸等元数据信息序列化存储于 JSON 配置文

件中，为后续的视觉动画准备结构化素材。

2. 内容生成与视听内容创建功能：核心内容自动化生成模块负责将静态内容转化为动态叙事。首先，系统调用大语言模型 API，基于预设的教学化提示词模板，将提取的 PPT 文本转化为符合口语习惯、结构清晰并与每页内容精确对应的讲解文案。随后，语音合成模块调用高质量 TTS 服务（如讯飞超拟人语音），将这些分页讲稿文本逐页转换为自然流畅、富有表现力的语音音频文件。

3. 视频动画合成与最终成片功能：动态视觉合成与后期处理模块是视听体验的构建关键。动态视频生成功能利用提取的背景图片和元素配置信息，通过 FFmpeg 为每一页幻灯片构建时间线动画，控制各个可视化元素按逻辑顺序和指定动画效果（如淡入、滑入）在背景上显现，生成无声的动态视频片段。音视频同步功能则将这些无声视频与对应的讲解音频进行精确的对齐与封装，确保语音讲解与视觉元素的动态变化实现毫秒级同步，输出带音频的独立视频片段。最终，所有独立的视频片段按照原 PPT 顺序，通过无缝拼接并辅以平滑的转场效果（如淡入淡出），合成为一个完整的教学视频。

4. 系统控制与流程管理功能：系统支撑与流程管控保障了流程的健壮性与易用性。系统需要提供清晰的启动入口，例如通过命令行或简易图形界面引导用户输入必要参数，启动自动化流程。在整个处理过程中，系统必须具备完善的异常捕获、处理与日志记录机制，确保当任一模块出现故障时，流程能够恰当中止，并为用户或维护者提供明确、可操作的错误诊断信息，便于快速定位和排查问题。

2.3 业务流程分析

核心业务流程是一条严格的单向线性流水线：

用户上传 PPT -> 系统解析 PPT（文本+图片+元素） -> AI 生成分页讲稿 -> TTS 合成分页音频 -> 基于元素生成分页动画视频 -> 将分页音频与动画视频同步合并 -> 将所有分页视频顺序拼接 -> 生成并交付最终视频。

每个环节的输出是下一环节的输入，依赖关系明确。流程不可逆，但应允许从中间某个环节（在提供必要中间文件的情况下）重新开始，以提高调试和重试效率。

2.4 非功能需求分析

1. 可靠性：系统可靠性是保障流程稳定运行的基础。系统需能稳健处理主流的 PPTX 文件格式，并对复杂的页面排版具备一定的容错能力。在调用外部 AI 与语音合成服务时，必须设计完善的网络超时处理与请求重试机制，以应对不稳定的网络环境或服务端短暂故障。整体架构应具备良好的容错性，单个功能模块的失败不应引致整个系统崩溃，而应能捕获异常、记录详尽的错误日志，并实现流程的优雅中止，为用户提供明确的故障定位信息。

2. 性能：在性能方面，对于一份 50 页以内的常规 PPT，全流程自动化处理时间应

力争控制在分钟级别（例如 30 分钟内），其耗时主要集中于外部 API 调用与本地视频渲染环节，此效率对目标用户而言具备可用性。同时，本地 FFmpeg 视频合成过程需合理管控 CPU 与内存的资源占用，避免对用户计算机造成过大负荷。在输出质量上，最终视频应至少达到 1080p 分辨率与 30fps 的流畅帧率，确保画质清晰。合成语音需自然流畅、接近真人发音，且与动态画面保持毫秒级的精确同步。所有视觉元素的动画效果应平滑自然，其出现节奏需与讲解语义相匹配，以达成最佳的视听学习效果。

3. 易用性、可维护性与成本控制：在易用性、可维护性与成本控制方面，系统需提供尽可能简化的操作界面，例如通过单一命令行或简易图形界面引导用户输入核心参数，降低使用门槛。架构设计必须遵循模块化、高内聚低耦合的原则，使得各功能模块（如解析、生成、合成等）能够独立测试、升级甚至替换（如切换不同的 TTS 服务提供商）。同时，系统应将 AI 提示词、语音参数、动画时序、输出格式等关键配置项外置于配置文件，实现灵活定制而无需修改源代码。此外，设计方案需充分考虑外部 API 调用的经济成本，通过优化提示词工程以降低 Token 消耗，或提供不同质量档位的语音合成选项，供用户根据实际需求在效果与成本之间进行权衡。

3. 设计和实现

3.1 体系结构设计

本项目采用模块化、流程驱动的体系结构设计，遵循“输入-处理-输出”的线性流水线模型。核心思想是将复杂的教学视频生成任务分解为一系列有序、独立的子任务，每个子任务由一个专门的功能模块负责，模块之间通过标准化的数据格式（如文本文件、JSON 配置文件、音频/视频文件）进行衔接。系统架构分为三层：数据输入层、核心处理层和输出生成层。

数据输入层以 PPT 文件为原始素材，负责解析并提取文本内容和视觉元素。核心处理层是系统的主体，依次执行 AI 讲稿生成、语音合成、PPT 元素分离、动画视频生成、音视频同步等关键转换操作。输出生成层负责整合中间产物，最终合成一个完整的、包含讲解音频和动态视觉效果的教学视频。这种分层、分阶段的架构确保了流程的清晰性、模块的可维护性，以及任务失败时的易排查性。

3.2 功能模块设计

根据主流程，系统具体划分为以下八个核心功能模块：

1.PPT 解析模块（extract_ppt_text，pptx_to_images）负责读取 PPTX 格式文件。其核心子模块用于提取每页幻灯片中的文本内容，供后续讲稿生成使用；另一子模块可将每页 PPT 转换为一张静态图片，作为视频生成的视觉基底。

2.AI 讲稿生成模块（generate_ai_script）：调用外部大语言模型 API（硅基流动），

将提取的 PPT 文本作为上下文提示，生成与每页 PPT 内容相匹配、适合口语化讲解的详细讲稿文本，并保存为结构化的文本文件。

3.语音合成模块（synthesize_voices）：调用语音合成 API（讯飞星火超拟人语音），将上一步生成的每页讲稿文本转换为对应的、自然流畅的语音音频文件（MP3 格式）。

4.PPT 元素提取模块（extract_only_images）：深度解析 PPT 文件，识别并分离出每页中的独立图形、图像等可视化元素，将其类型、位置、尺寸等信息序列化保存至 JSON 配置文件（extract_pic.json）。这一步是为创建动态动画准备素材。

5.背景图生成模块（run_deletion_test）：利用上一步生成的 JSON 配置，从原始的 PPT 页面图片中移除所有可分离的可视元素，生成干净的、仅包含背景和不可编辑文本的底层背景图片，存储于/img 目录。

6.单页动画视频生成模块（generate_all_ppt_videos）：基于背景图片和 JSON 中的元素信息，利用 FFmpeg 图像处理库，为每一页 PPT 生成一段动态视频。视频中，图片元素依次在背景上出现，模拟讲解时的视觉引导效果。

7.音视频合成模块（merge_video_audio）：将每页对应的动画视频与讲解音频进行对齐和合并，确保画面切换与语音讲解同步，生成一系列带音频的单页视频片段。

8.最终视频合并模块（merge_videos）：使用 FFmpeg 工具，将所有带音频的单页视频片段按 PPT 原顺序无缝连接并为每个单视频添加渐隐渐显效果，合成为一个完整的、连贯的教学视频文件。

3.3 核心技术设计

1.PPT 处理技术：采用 python-pptx 库进行 PPT 文件的底层解析，实现对幻灯片、形状、文本框及图片元素的精准访问与属性读取。

2.AI 集成技术：通过 HTTP 请求（使用 requests 库）调用硅基流动 API。设计特定的提示词工程模板，将 PPT 文本作为输入，引导 AI 生成结构完整、语言生动、符合教学场景的讲解脚本。

3.语音合成技术：集成讯飞星火语音合成 API，通过 webSocket 连接，提交文本并接收音频流。关键技术点在于处理权限认证、参数配置（如发音人、语速、音调）以及音频数据的接收与本地保存。

4. 动态视觉生成技术：核心依赖于 FFmpeg 的多媒体处理能力。通过编写复杂的

FFmpeg 命令或使用 ffmpeg-python 等封装库，实现：

- **图像序列处理**：将静态图片转换为视频流。
- **元素动画**：利用 filter_complex 功能，基于时间线为每个提取的元素添加位置移动、透明度变化等动画效果。
- **音视频对齐与混合**：确保音频时长与视频时长匹配，并将它们封装到同一容器中。

5.流程控制与数据持久化：使用 Python 进行整体流程编排。利用 JSON 格式作为模块间数据交换的标准，保存元素的空间信息和动画参数。通过严格的异常捕获和错误处理，确保单个步骤的失败不会导致整个流程崩溃，并提供明确的错误日志。

3.4 开发环境

- 编程语言：Python 3.8+
- 核心依赖库：
  - python-pptx：用于 PPT 文件解析。
  - requests：用于调用硅基流动、讯飞星火等外部 HTTP API。
  - ffmpeg-python 或 subprocess：用于封装和调用 FFmpeg 命令行工具。
  - Pillow (PIL)：用于基础的图像处理操作（如需要）。
  - json：用于配置文件读写。
- 关键外部工具：
  - FFmpeg：必须预先安装并配置在系统环境变量中，是本项目视频处理的核心引擎。
  - 硅基流动 API 账户：用于获取 AI 讲稿生成服务。
  - 讯飞星火 API 账户：用于获取超拟人语音合成服务。
- 开发环境：可在 Windows、macOS 或 Linux 系统上开发，需确保 Python 环境和 FFmpeg 的兼容性。

3.5 主要功能实现

主函数 main 清晰实现了上述模块的串联执行流程：

1.解析 PPT：调用 extract_ppt_text 函数，遍历 PPT 所有幻灯片，提取文本框中的文字，合并为连贯的文本字符串，为 AI 生成提供素材。

2.AI 生成讲稿：调用 generate_ai_script 函数，将上一步得到的 PPT 文本发送给硅基流动 API。函数内部设计提示词，要求 AI 根据 PPT 内容扩展出讲解词，并将返回的结

果按页分割保存。

3.语音合成：调用 synthesize_voices 函数，遍历上一步生成的每页讲稿文本文件，调用讯飞星火 TTS API，生成对应的语音文件，并按页编号存储。

4.提取 PPT 图片元素 调用 extract_only_images 函数,使用 python-pptx 分析 PPT 中每个形状，筛选出图片类型或特定图形，将其属性（如图片数据、位置）导出到 extract_pic.json 文件。

5.生成纯净背景图：调用 run_deletion_test 函数，读取 JSON 文件，通过图像处理技术（例如，利用元素位置信息在原图上进行覆盖或裁剪）从每页 PPT 渲染图（或通过 pptx_to_images生成的图）中移除可提取元素，生成仅剩背景和基础文字的图片，存入 /img 文件夹。

6.生成单页动画视频:调用 generate_all_ppt_videos 函数，为每一页 PPT 执行以下操作：加载背景图；根据 JSON 文件加载元素图片；使用 FFmpeg 构建复杂的滤镜图，将背景设为轨道，并将元素作为叠加层，为每个叠加层设定出现时间、持续时间和动画效果（如从左侧滑入）；渲染输出一段无声的动画视频片段。

7.合成带音频单页视频：调用 merge_video_audio 函数，对于每一页，使用 FFmpeg 的-i 参数同时输入该页的无声动画视频和对应的讲解音频文件，将它们合并封装为一个新的视频文件，确保音画同步。

8.合并最终视频：调用 merge_videos 函数，使用 FFmpeg 的 concat 协议或滤镜，按顺序列表将所有带音频的单页视频文件连接起来，生成一个最终的教学视频输出文件（如 final_output.mp4）。

整个实现过程通过 Python 脚本将 AI 能力、云计算服务与强大的本地多媒体处理工具链相结合，实现了从静态 PPT 到动态讲解视频的自动化生产流水线。

4.系统测试

4.1 功能测试

4.1.1PPT 解析模块测试

测试用例：输入标准 PPTX 格式文件
测试结果：

成功提取 PPT 中所有文本内容，格式化为"第 n 页：文字内容"

准确将每页 PPT 转换为 PNG 格式图片，保存至 img 目录

支持复杂 PPT 结构，包括组合形状和表格文本提取
图片输出分辨率可调，默认 96DPI 保证清晰度

4.1.2AI 讲稿生成模块测试

测试用例：调用硅基流动 API 生成讲稿
测试结果：

API 调用成功，返回格式符合预期要求
讲稿内容自然流畅，每页控制在 50 字以内
成功验证返回格式并提取每页讲稿保存为单独文件
具备错误处理机制，API 异常时优雅降级

4.1.3 语音合成模块测试

测试用例：使用讯飞 TTS 合成语音
测试结果：

WebSocket 连接稳定，认证机制正常
支持多种发音人选择，语音质量自然
音频文件按页编号保存，格式为 MP3
具备网络异常重试机制，合成失败有明确提示

4.1.4PPT 元素提取模块测试

测试用例：深度解析 PPT 元素结构
测试结果：

准确识别图片元素，过滤文本框和形状
提取元素坐标、尺寸等元数据，支持 EMU 和像素单位转换
成功保存图片到本地，并生成结构化 JSON 配置文件
处理复杂 PPT 布局，支持嵌套元素识别

4.1.5 背景图生成模块测试

测试用例：生成纯净背景图片
测试结果：

基于 XML 精准删除指定元素，保留背景内容
生成的背景图片质量清晰，元素移除准确
支持批量处理，自动保存到指定目录
错误处理完善，文件不存在时友好提示

4.1.6 单页动画视频生成模块测试

测试用例：创建元素动画视频
测试结果：
元素按顺序动态出现，动画效果平滑自然
支持自定义元素停留时间，默认 18 秒/元素
背景与元素比例自适应，保持原始布局
FFmpeg 合成成功，视频格式标准兼容

4.1.7 音视频合成模块测试

测试用例：视频音频同步合并
测试结果：
智能处理音视频时长差异，音频长时自动延长视频
支持两种延长方案：tpad 滤镜和 concat 协议
音画同步精确，输出视频质量稳定
具备完善的错误处理和日志记录

4.1.8 最终视频合并模块测试

测试用例：多视频片段拼接
测试结果：
成功按页码顺序拼接所有视频片段
添加渐入渐出转场效果，过渡自然
支持各种视频格式，输出为标准 MP4
文件大小优化，1080p 分辨率保证清晰度

5. 总结与展望

5.1 总结

本项目成功实现了从静态 PPT 到动态教学视频的全自动生成系统，主要成果包括：

5.1.1 技术成果

完整的自动化流水线：
构建了包含 8 个核心模块的端到端解决方案，实现 PPT 解析、内容生成、视听合成

的一站式处理。
深度集成 AI 能力：
创新性地结合大语言模型的内容生成和高质量语音合成技术，赋予静态内容动态表达能力。
精准的元素级控制：
通过深度解析 PPT 结构，实现可视化元素的精确提取和动画控制，超越传统模板化方案。
稳健的工程实现：
采用模块化架构，具备完善的错误处理和日志记录，保证系统可靠运行。

5.1.2 应用价值

显著降低制作门槛：
将专业的视频制作技术封装为自动化工具，使非专业用户也能快速生成高质量教学视频。
大幅提升制作效率：
传统需要数小时的手工视频制作，现在仅需几十分钟即可完成，效率提升 10 倍以上。
保证内容质量一致性：
自动化流程确保每页内容的处理标准和视觉效果统一，避免人工操作差异。
支持个性化定制：通过配置参数调整，满足不同用户的风格偏好和制作需求。

5.1.3 技术创新点

多技术栈融合：
将文档处理、AI 生成、语音合成、视频编辑等异构技术有机整合。
智能时长适配：
创新的音视频时长智能匹配算法，确保内容同步自然。
跨平台兼容性：
基于 Python 和 FFmpeg，支持 Windows、macOS、Linux 等多平台部署。

5.2 展望

尽管本项目已实现预期目标，但仍有多方面可进一步优化和完善：

5.2.1 技术深化方向

增强 AI 理解能力：
引入多模态大模型，使系统能够"看懂"PPT 中的图表、公式等复杂内容，生成更精

准的讲解。
丰富动画效果库：
开发更多样化的元素动画效果，支持用户自定义动画轨迹和时序。
智能节奏控制：
基于内容重要性自动调整讲解节奏和元素呈现速度。
实时预览编辑：
开发图形化界面，支持用户实时调整生成效果。

5.2.2 功能扩展方向

多语言支持：
扩展支持英语、日语等多语言 PPT 的讲解生成。
多发音人选择：
提供更多音色和风格的语音合成选项。
模板主题系统：
开发不同学科风格的主题模板，如理科、文科、商务等。
互动元素支持：
生成包含测验、问答等互动环节的视频内容。
工程优化方向
分布式处理：
将视频生成任务分布到多台机器并行处理，进一步提升效率。
云服务部署：
提供 Web 版服务，用户无需安装本地环境即可使用。
API 开放平台：
将核心能力封装为 API，支持第三方应用集成。
质量评估体系：
建立自动化的视频质量评估指标，优化生成效果。

5.2.3 生态建设方向

教育资源共享：
与在线教育平台合作，构建教学视频资源库。
企业培训应用：
拓展到企业内训、产品演示等商业场景。
个性化学习：
结合学习者画像，生成个性化的学习内容。

参考文献

[1]FutureUniant. (2025). Tailor[Computer software]. GitHub. https://github.com/FutureUniant/Tailor
[2]galis. (2025). OpenTikTok[Computer software]. GitHub. https://github.com/galis/OpenTikTok
[3]LumingMelody. (2025). Ai-movie-clip[Computer software]. GitHub. https://github.com/LumingMelody/Ai-movie-clip
[4]Zhu, Z., Lin, K. Q., & Shou, M. Z. (2025). Paper2Video: Automatic Video Generation from Scientific Papers. arXiv preprint. https://arxiv.org/abs/2510.05096