「AI多感官智能閱讀器」

# 1. 影像辨識子系統 (Image Recognition Subsystem)

* 核心目標： 快速、準確地從使用者提供的影像中辨識出書本的唯一身份（如ISBN或書名+作者）。

## 1-1.主要功能/模組：

### 影像擷取模組： 負責啟動裝置鏡頭，提供穩定、清晰的即時影像流給使用者預覽。

1. 影像預處理模組： 在進行辨識前，自動對擷取的影像進行優化，包含：
2. 邊緣偵測與裁切：自動找到書本封面的範圍，排除背景干擾。
3. 亮度與對比度校正：應對不同環境光源的影響。
4. 透視校正：修正因傾斜拍攝造成的封面變形。

## 1-2.特徵辨識與比對模組：

1. 方案A (雲端API)：將預處理後的影像傳送至成熟的視覺辨識服務（如 Google Vision API），利用其強大的模型進行辨識，返回可能的書名或關鍵字。
2. 方案B (本地端模型)：在App中部署輕量化的機器學習模型（如 TensorFlow Lite），進行離線辨識。此方案速度快，但模型需要預先訓練，準確度可能受限。

## 1-3.技術挑戰：

如何處理反光、陰影、模糊或部分遮擋的封面。

同一本書不同版本（新版、舊版、紀念版）封面的辨識問題。

辨識速度與準確度之間的平衡。

# 2. AI內容生成子系統 (AI Content Generation Subsystem)

* 核心目標： 基於辨識出的書本資訊，動態生成高品質、結構化、且富有吸引力的導讀內容。

2-1.主要功能/模組：  
 圖書資料擴充模組：

1. 利用辨識結果（ISBN），透過API介接外部圖書資料庫（如 Google Books API, OpenLibrary API），獲取書籍的詳細元數據（Metadata），包含摘要、作者簡介、出版日期、類型、評論等。
2. 提示詞工程模組 (Prompt Engineering)： 這是整個系統的「靈魂」。根據書籍的元數據和使用者需求，設計精巧的指令（Prompt），引導大型語言模型（LLM）生成特定風格和結構的內容。例如：

* Prompt 1 (摘要用): "請用300字、活潑的說書人語氣，總結這本書的核心論點..."
* Prompt 2 (深度分析用): "針對書中的主角，分析其性格轉變的三个關鍵事件..."
* LLM介接模組： 負責與大型語言模型（如 Google Gemini API 或 OpenAI GPT API）進行穩定、高效的通訊，發送請求並接收生成的內容。
* 內容結構化模組： 將LLM返回的純文字內容，進行解析和格式化，拆分成結構化的資料（如JSON），包含：{ "語音稿": "...", "關鍵字": ["...", "..."], "建議配圖": "...", "建議配樂風格": "..." }。

### 2-2. 技術挑戰：

1. 如何設計有效的提示詞以確保生成內容的準確性、避免AI產生幻覺（Hallucination）。
2. 多語言內容生成的品質控制。
3. API呼叫的成本與延遲管理。

3. 多感官輸出子系統   
(Multi-sensory Output Subsystem)

* 核心目標： 將結構化的內容腳本，轉化為同步且流暢的聽覺與視覺體驗。

3-1.主要功能/模組：

* 語音合成模組 (TTS)：

1. 將「語音稿」文字轉換為自然、帶有情感的人聲。
2. 需支援多種語言和聲音風格（如知性女聲、沉穩男聲）的切換。

* 視覺化引擎：

1. 負責在播放介面上，根據時間軸同步顯示「關鍵字」、「字幕」。
2. 根據「建議配圖」的描述，透過API從圖庫（如 Unsplash）或AI繪圖服務抓取或生成對應的視覺素材。

### 音訊混音器：

將TTS生成的人聲、背景音樂（BGM）、情境音效（SFX）三條音軌進行混音 和同步播放，營造沉浸感。

3-2.技術挑戰：

1. 確保語音、字幕、圖像、音效在時間軸上的精準同步。
2. 尋找或生成高品質且版權合規的音樂與圖像素材。
3. TTS的自然度與情感表現力。  
     
     
     
   4. 使用者介面與控制子系統   
   (UI & Control Subsystem)

核心目標： 提供一個極簡、直觀的操作介面，讓使用者能輕鬆完成從掃描到體驗的全過程。

4-1. 主要功能/模組：

1. 掃描介面： 簡潔的相機預覽畫面，引導使用者將書本封面置於框內。
2. 播放介面： 整合視覺化引擎的輸出，提供播放/暫停、進度條、音量控制等基本功能。
3. 互動選項： 在導讀結束後，提供「深入探索」、「重聽一次」、「更換語言」等選項，讓使用者能進行下一步操作。
4. 設定介面： 允許使用者選擇偏好的語音、字幕大小等個人化設定。

## 4-2.技術挑戰：

1. 跨平台（iOS/Android）的UI一致性與性能。
2. 如何在功能豐富與介面簡潔之間取得平衡，避免使用者感到混亂。