

貼圖辨識來源擴充(圖檔貼上 + 10MB 縮圖修正)

Ctrl+V 支援複製的圖檔(CF_HDROP)，並修正大圖超過 API 10MB 上限 (2026-06-06)

一、訴求與現象

原本 Ctrl+V

只認剪貼簿的「點陣圖」(截圖)。希望在檔案總管對一張圖檔(.jpg/.png 等)按 Ctrl+C 後，於 STK32 Ctrl+V 也能直接辨識。實作後測試大張 jpg 時又出現「無法解析」。

二、踩坑與陷阱

坑	現象 / 根因	解法
複製圖檔貼不上	檔案總管 Ctrl+C 圖檔時，剪貼簿是 CF_HDROP(檔案清單)而非 CF_DIB(點陣圖)，原程式只認 CF_DIB。	gtwvt.c 加 CF_HDROP 分支:手動解析 DROPFILES 取第一個檔，副檔名是影像就複製成 stk_clip.bmp。
中文檔名	DROPFILES 可能是寬字元(fWide)，ANSI 取檔名會亂碼。	fWide 時用 WideCharToMultiByte(CP_ACP) 轉碼後再 CopyFile。
C 字串反斜線	Python bytes 寫 C 字串時 \\ 被吃成單一反斜線 → C 變無效跳脫、路徑壞掉。	以顯式 \x5c\x5c 寫入，對齊既有正確寫法。
大圖 API 400	2.7MB jpg 經 PIL 轉 PNG 後膨脹成 14MB，超過 Anthropic 10MB 上限 → HTTP 400「image exceeds 10 MB」。	img_to_png_b64 轉檔前先縮到長邊 1568px(Anthropic 建議值)，必要時再逐步縮，維持 PNG。

三、解決方法

- C 端(gtwvt.c):新增 hb_gt_wvt_ClipboardFileToBmp()，沒有 CF_DIB 時改檢查 CF_HDROP，把複製的圖檔複製成 %TEMP%\stk_clip.bmp，再走原本的 9002 流程。img2text 用 PIL 依「內容」判圖，所以即使副檔名是 .bmp 也能正確讀入 jpg/png。
- Python 端(img2text.py):img_to_png_b64 加入長邊 1568px 縮圖，徹底解決 10MB 超限。

四、使用技術與用法

技術	用法 / 位置
CF_HDROP 解析	自定 DROPFILES 結構(免相依 shell32 DragQueryFile)，GlobalLock 後取 pFiles 偏移第一個檔。
影像副檔名判 斷	lstrcmpiA 比對 .jpg/.jpeg/.png/.bmp/.gif/.webp/.tif/.tiff。
圖檔複製	CopyFileA(來源, %TEMP%\stk_clip.bmp)；位置:gtwvt_patch\gtwvt.c。
縮圖	PIL im.resize / thumbnail，長邊 1568px，LANCZOS；位置:img2text.py img_to_png_b64()。
重建	C 改動跑 stkg_tlib\build_StkGTWVT.bat;img2text 用 PyInstaller 32-bit(E:\Python312-32)重編後複製到 HBSTK\build。

五、使用方式

檔案總管對名片圖檔按 Ctrl+C → 回 STK32

Ctrl+V，後續流程與貼點陣圖完全相同。img2text 是獨立 process，只更新
img2text.exe 即可，不必重編 Stk32。