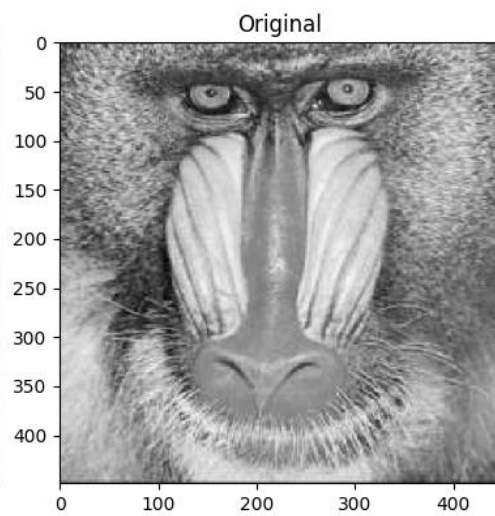
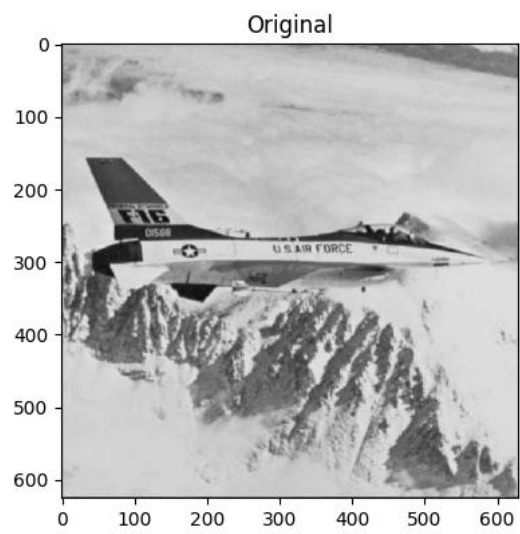
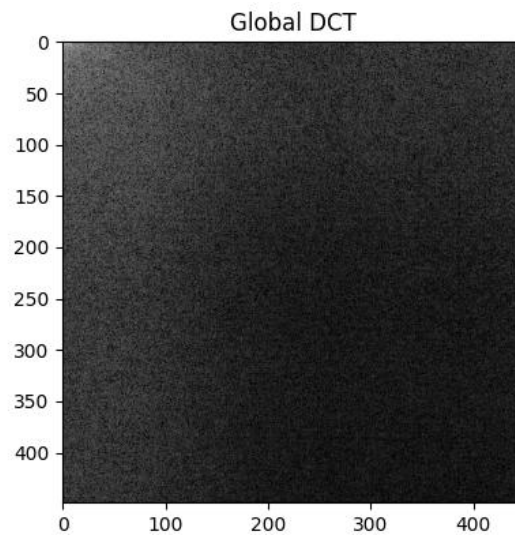
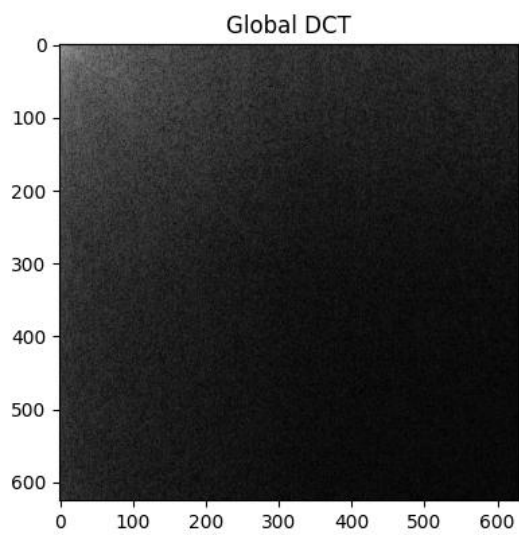


1.

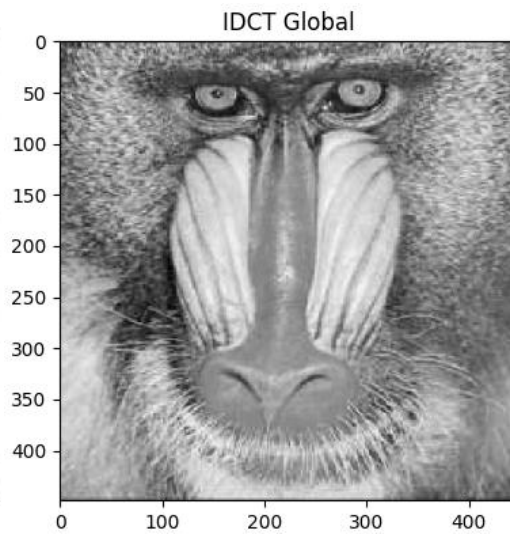
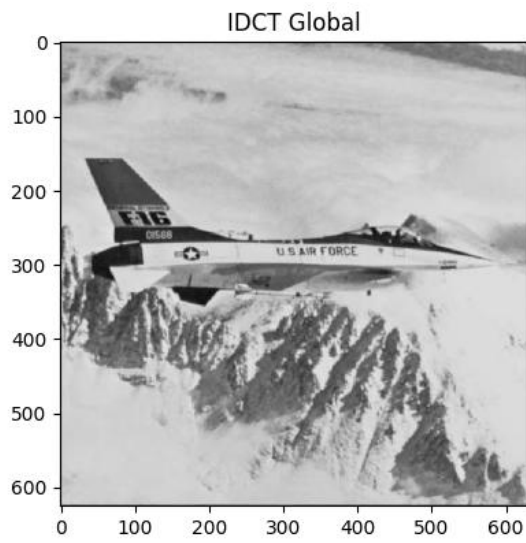
Origin image



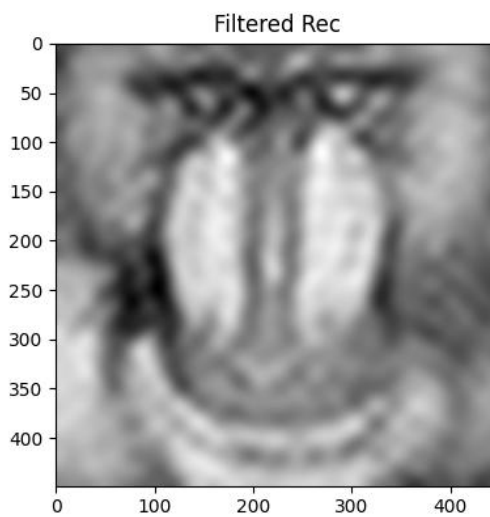
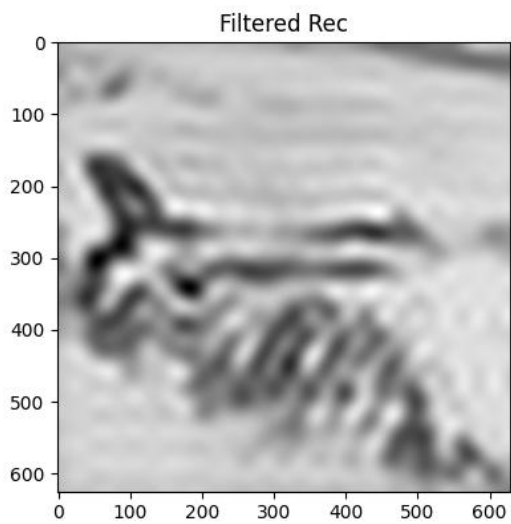
Global DCT Frequency Domain



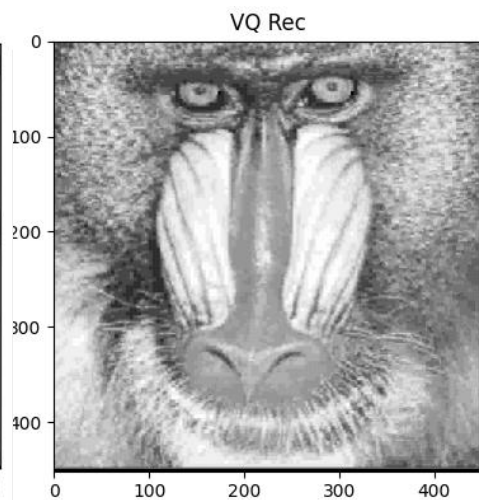
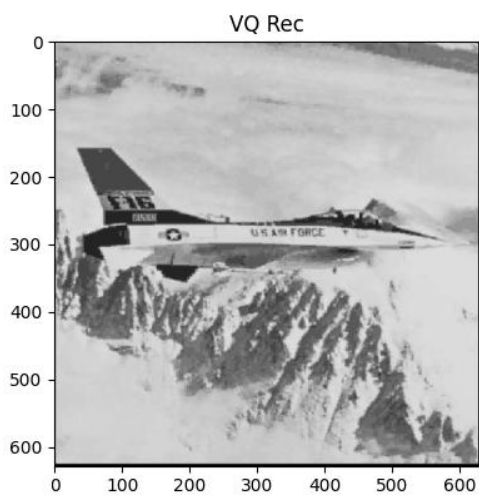
IDCT Reconstruction



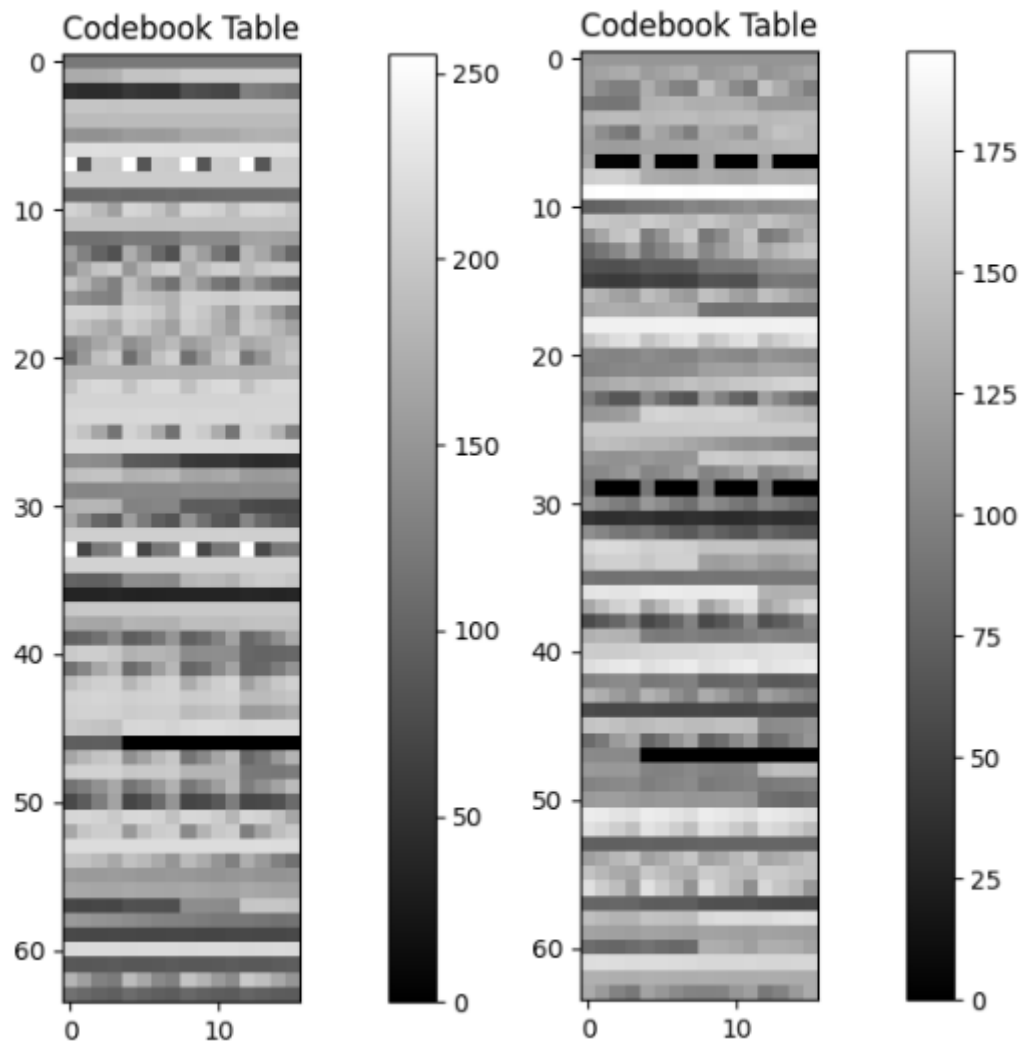
Low-frequency Cropping



VQ Reconstruction



Codebook Table



2.

在 DCT 部分，我先對整張影像進行全域二維 DCT，觀察空間域如何轉換成頻域；接著進行 8×8 區塊式 DCT 模擬 JPEG 的運作方式。之後再進行頻域濾波，只保留左上角低頻區域，其餘係數設為 0，並透過 IDCT 還原，以評估經過低頻壓縮後的影像品質。

在 VQ 部分，我將影像分割成不重疊的 4×4 區塊並展平成 16 維向量，利用所有區塊向量透過 LBG 演算法訓練出 64 個 codeword 的 codebook。編碼階段將每個區塊對應到最接近的 codeword，解碼則以 codeword 取代原始區塊重建影像。最後將 DCT 與 VQ 的視覺結果比較，以分析兩者的壓縮特性。

3.

在做 DCT 的時候，我最明顯的感受就是：全域 DCT 的還原效果真的比區塊式 DCT 好很多。當我改用 8×8 block 之後，只要我把高頻係數砍得比較兇，影像就會開始出現一格一格的方塊狀失真，跟 JPEG 壓縮過度時的那種破碎感幾乎一模一樣。這讓我蠻直觀地理解到為什麼 JPEG 在高壓縮比會變成那種畫面。

另外，我也發現低頻訊息真的超重要。就算高頻幾乎被砍光，影像還是看得出主要輪廓，這讓我更確定人眼主要靠低頻來辨識大結構，高頻反而只是補細節。

做到 VQ 的時候，我也遇到另一種特別的視覺效果：重建出來會有「拼貼感」或像「卡通濾鏡」一樣。原因很明顯，就是 codebook 的數量有限，所以很多不同的區塊都被量化成同一個 codeword。這也讓我體會到 VQ 跟 DCT 不太一樣，VQ 的效果完全吃 codebook 的質量，而 LBG 在資料量大時收斂又慢，初始化也影響結果，整體來說比較不穩定。